VŠB-Technical university of Ostrava
Faculty of Electrical Engineering and Computer Science
Department of Applied Mathematics

Scalable algorithms for solving elasto-plastic problems

2012

Martin Čermák

First of all I would like to thank my supervisor doc. Ing. Tomáš Kozubek, Ph.D. for his patience, help, support, and leadership during my doctoral studies, also many thanks to Mgr. Stanislav Sysala, Ph.D. for helping me with the beginning of study of elasto-plastic problems, leadership through the theoretical background, reading many versions of my thesis and making corrections, and for many very interesting discussions not only about mathematics. Thanks to Ing. Alexandros Markopoulos with helping me with the implementation to MatSol library. Also, I would like to thank all members of the Department of Applied Mathematics for a possibility to be a part of the team. And last but not least, I thank my girlfriend, my parents, and friends, which supported me for the whole time of my studies, for everything they have done for me.

This research was supported by the grant of the GA CR 103/09/H078.

This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the project SPOMECH - Creating a multidisciplinary R&D team for reliable solution of mechanical problems, reg. no. CZ.1.07/2.3.00/20.0070 within Operational Programme 'Education for competitiveness' funded by Structural Funds of the European Union and state budget of the Czech Republic.

Abstrakt

V této disertační práci představíme algoritmy pro efektivní paralelní implementaci elastoplastických problémů se zpevněním, založených na metodě rozložení oblastí zvané TFETI (Total Finite Element Tearing and Interconnecting). Uvažujeme tři odlišné asociativní elastoplastické modely: von Misesův model s izotropním zpevněním, von Misesův model s kinematickým zpevněním a Drucker-Prager perfektně plastický model. Tyto modely jsou diskretizovány v čase pomocí implicitní Eulerovy metody a každý časový krok elastoplastického modelu je diskretizován v prostoru pomocí metody konečných prvků. Tímto dostaneme pro každý časový krok systém nelineárních rovnic pro rovnostní omezení nebo systém nelineárních nerovnic se silně nehladkým a silně monotónním operátorem. Nehladká Newtonova metoda se aplikuje na tento nelineární systém za účelem jeho linearizace. Odpovídající linearizované problémy vyskytující se v Newtonových iteracích jsou paralelně nebo sekvenčně řešeny pomocí metody TFETI. Zmíněné algoritmy založené na TFETI byly implementovány v Matlabovském paralením prostředí a jejich funkčnost byla ilustrovaná na 2D a 3D elastoplastických příkladech. Numerické výsledky prezentujeme pro různé časové diskretizace a různé konečno-prvkové sítě, pozorujeme kvadratickou konvergenci nehladké Newtonovy metody. Také demonstrujeme paralelní a numerickou škálovatelnost těchto algoritmů na řešení úloh elastoplasticity.

Klíčová slova: Elastoplasticita, Total-FETI, izotropní zpevnění, kinematické zpevnění, Nehladká Newtonova medoda, PCGP, SMALSE-M, von Misesův elastoplastický model, Drucker-Prager elastoplastický model

Abstract

In the thesis we propose an algorithm for the efficient parallel implementation of elastoplastic problems with hardening based on the so-called TFETI (Total Finite Element Tearing and Interconnecting) domain decomposition method. We consider three different associated elasto-plastic models: the von Mises model with isotropic hardening, the von Mises model with kinematic hardening and the Drucker-Prager perfectly plastic model. Such models are discretized by the implicit Euler method in time and the consequent one time step elasto-plastic problem by the finite element method in space. The latter results in a system of nonlinear equations for equality constraints or system of nonlinear inequations with a strongly semismooth and strongly monotone operator. The semismooth Newton method is applied to solve this nonlinear system. Corresponding linearized problems arising in the Newton iterations are solved in parallel or sequentially by the above mentioned TFETI domain decomposition method. The proposed TFETI based algorithm

was implemented in Matlab parallel environment and its performance is illustrated on 2D and 3D elasto-plastic benchmarks. Numerical results for different time discretizations and mesh levels are presented and discussed and a local quadratic convergence of the semismooth Newton method is observed. We also demonstrate parallel and numerical scalability of the proposed algorithms for solving elasto-plasticity.

Keywords: Elasto-plasticity, Total-FETI domain decomposition, isotropic hardening, kinematic hardening, semismooth Newton method, PCGP, SMALSE-M, von Mises elasto-plastic criterion, Drucker-Prager elasto-plastic criterion

List of shortcuts and symbols

$\mathbb{R}^3$	-	three-dimensional real-valued space
Ω, Ω^d, Ω^o	-	domain represented a body
Ω^p, Ω^s, Ω^{s_d}	-	subdomain
o	-	number of bodies
s	-	number of subdomains Ω
Γ	-	boundary of Ω
Γ_U	-	part of Γ where the Dirichlet boundary conditions are prescribed
Γ_N	-	part of Γ where the Neumann boundary conditions are prescribed
Γ_C	-	part of Γ where the contact boundary conditions are prescribed
$x = (x_1, x_2, x_3)$	-	spatial variable, in numerical examples we use also convention (x, y, z)
t	-	time variable
u	-	displacement
v	-	test function
g	-	volume forces
n	-	outward unit normal tensor
F	-	surface forces
σ	-	Cauchy stress tensor
ε	-	small strain tensor
ε^e	-	elastic part of the strain tensor ε
ε^p	-	plastic part of the strain tensor ε
κ	-	isotropic hardening variable
V	-	space of kinematically admissible displacements
V_h	-	subspace of V, with piecewise linear and continuous functions
S	-	space of all symmetric second order tensors in 3D
$L^2(\Omega),\ H^1(\Omega)$	-	Hilbert spaces
$\langle ., . \rangle_F$	-	Frobenius scalar product
$\|.\|_F$	-	Frobenius norm
$\mathbb{C}$	-	Hooke fourth order tensor
$\mathbb{C}^{-1}$	-	inverse of the Hooke fourth order tensor
λ, μ	-	Lame coefficients
E	-	Young modulus
ν	-	Poisson ratio
$\mathrm{vol}(\eta)$	-	volumetric part of a tensor η
$\mathrm{dev}(\eta)$	-	deviatoric part of a tensor η
$\mathrm{tr}(\eta)$	-	trace of a tensor η
I	-	unit second order tensor
$\mathbb{I}_d$	-	fourth order tensor representing the deviatoric part
$\Phi(\sigma, \kappa)$	-	yield function for the von Mises criterion and isotropic hardening
H_m	-	isotropic hardening modulus
σ_y	-	yield stress
$i,\ k$	-	indices (Newton, time)

$\triangle t_{k+1}$	-	increment of time $\Delta t_{k+1} = t_{k+1} - t_k$
$\triangle \varepsilon_{k+1}$	-	time increment of the strain $\Delta \varepsilon_{k+1} = \varepsilon_{k+1} - \varepsilon_k$
$\hat{n}(\sigma_k)$	-	deviatoric part of σ_k divided by its norm
$T_\sigma(\tau, \omega; .)$	-	stress operator
$T_\kappa(\tau, \omega; .)$	-	hardening operator
Φ^+	-	positive part of the function Φ
$T_k(.)$	-	stress operator $T_\sigma(\sigma_k, \kappa_k; .)$ with respect to the current parameters σ_k, κ_k
T_k^o	-	Clark generalized derivative of T_k
$\mathcal{T}_h$	-	shape regular triangulation of Ω
T	-	local element, $T \in \mathcal{T}_h$
$\mathcal{N}$	-	number of vertices of triangulation $\mathcal{T}_h$
n	-	number of DOFs (degrees of freedom), $n = 3\mathcal{N}$
$\boldsymbol{B}_U$	-	matrix representing the homogenous Dirichlet boundary condition
$\mathbf{B}_G$	-	matrix representing the gluing condition
$\boldsymbol{u}$	-	displacement vector
$\boldsymbol{o}$	-	zero vector representing
$\boldsymbol{R}_T$	-	restriction operator of a displacement vector $\boldsymbol{u}$ to T
$\boldsymbol{u}_T$	-	displacement vector for local element T
$\boldsymbol{\varepsilon}$, $\boldsymbol{\sigma}$, $\boldsymbol{\kappa}$	-	vector representations of tensors ε, σ, and κ
$\boldsymbol{C}$	-	Hooke matrix, or the algebraic representation of the Hooke tensor $\mathbb{C}$
$\boldsymbol{E}_\sigma$	-	algebraic representation of the deviatoric operator $\mathbb{I}_d$ related to stress
$\boldsymbol{E}_\varepsilon$	-	algebraic representation of the deviatoric operator $\mathbb{I}_d$ related to strain
$\boldsymbol{\Phi}$, $\hat{\boldsymbol{n}}$	-	algebraic representation of the functions Φ, $\hat{n}$
$\boldsymbol{T}_{\kappa,k,T}$	-	algebraic representation of the function T_κ for constant stress and hardening
$\boldsymbol{T}_{k,T}$, $\boldsymbol{T}_{k,T}^o$	-	algebraic representations of the restrictions of the functions $T_{k,h}$, $T_{k,h}^o$
$\boldsymbol{G}$	-	matrix representing the algebraic relation between the strain and the displacement
$\boldsymbol{F}_k$	-	non-linear operator of one time step problem
$\boldsymbol{K}_k$	-	stiffness matrix which represents the bilinear form
$\boldsymbol{V}$	-	algebraic representation of space V_h
ϵ	-	value of stopping criteria
$\tilde{\boldsymbol{K}}$	-	restriction of $\boldsymbol{K}$ with respect to the Dirichlet conditions
$\boldsymbol{J}(\boldsymbol{v})$	-	functional defined on $\boldsymbol{v}$
V_h^p	-	space of piecewise linear and continuous approximations of $H^1(\Omega^p)$
$\mathtt{V}_h$	-	product space of V_h^p
$\mathtt{K}_h$	-	space of allowable displacement, which satisfy gluing conditions
$Ker\,\mathbf{K}$	-	kernel of matrix $\mathbf{K}$
$Im\,\mathbf{K}$	-	image of matrix $\mathbf{K}$
$\mathbf{R}$	-	matrix described the kernel of matrix $\mathbf{K}$
$\mathbf{K}^\dagger$	-	general pseudoinverse of matrix $\mathbf{K}$

$\mathbf{B}_E$	-	matrix representing equality constraints
$\mathbf{B}_I$	-	matrix representing inequality constraints
$\mathbf{B}$	-	matrix representing (all) constraints
$\mathbf{c}_I$	-	vector representing inequality constraints
$\mathbf{c}$	-	vector representing (all) constraints
$\boldsymbol{\lambda}$	-	Lagrange multipliers
$\boldsymbol{\alpha}$	-	amplitudes of rigid body motions
$\mathbf{F}$	-	substitutions matrix for $\mathbf{B}\mathbf{K}^{\dagger}\mathbf{B}^T$
$\mathbf{N}$	-	substitutions matrix for $-\mathbf{R}^T\mathbf{B}^T$
$\mathbf{d}$	-	substitutions vector for $\mathbf{B}\mathbf{K}^{\dagger}\mathbf{f}$
$\mathbf{e}$	-	substitutions vector for $-\mathbf{R}^T\mathbf{f}$
$\mathbf{P}_\mathbf{N}$	-	orthogonal projector onto $Ker\,\mathbf{N}$
m_U	-	number of mesh nodes with imposed Dirichlet boundary condition
$\mathtt{m}$	-	number of dual variables
$\mathtt{n}$	-	number of primal variables
l	-	dimension of null space

Contents

List of Tables

List of Figures

Introduction

The solvers for elasticity problems either with classical or contact boundary conditions are developed at our department for many years. These algorithms are implemented in MatSol library [45] and they are based on the Total-FETI (Total Finite Element Tearing and Interconnecting) domain decomposition method. They show both numerical and parallel scalability. The main goal of this thesis is a suitable combination of these algorithms with the semismooth Newton method for the solution of elasto-plastic problems either with classical or contact boundary conditions. This combination should extend the capabilities of the Matsol library to solve real-world problems in mechanics.

A linear solver used to solve linearized problems arising in each Newton's iteration is based on a FETI type domain decomposition method enabling the efficient parallel implementation. The standard FETI method (FETI-1) was originally introduced by Farhat and Roux [28]. Using this approach, a body is partitioned into non-overlapping subdomains, an elliptic problem with Neumann boundary conditions is defined for each subdomain, and intersubdomain field continuity is enforced via Lagrange multipliers. The Lagrange multipliers are evaluated by solving a relatively well conditioned dual problem of a small size that may be efficiently solved by a suitable variant of the conjugate gradient algorithm. The first practical implementations exploited only the favorable distribution of the spectrum of the matrix of the smaller problem [54], known also as the dual Schur complement matrix, but such algorithm was efficient only with a small number of subdomains. Later, Farhat, Mandel, and Roux introduced a "natural coarse problem" whose solution was implemented by auxiliary projectors and worked as preconditioning so that the resulting algorithm became in a sense optimal [29, 55].

In our implementation, we use the Total-FETI (TFETI) [21] variant of the FETI domain decomposition method, where even the Dirichlet boundary conditions are enforced by Lagrange multipliers. Hence all subdomain stiffness matrices are singular with a-priori known kernels which is a great advantage in the numerical solution. With known kernel basis we can regularize effectively the stiffness matrix without extra fill in and use any standard sparse Cholesky type decomposition method for nonsingular matrices [13]. To find the unknown part of dual solution we use the PCGP algorithm [29] which is known to be the effective solver for TFETI method.

The duality based domain decomposition methods may also be successful for the solution of the frictionless contact problems. The first observation was that the duality not only reduces the dimension and improves conditioning of the original problem, but it also reduces all the inequalities to the bounds on variables, so that the dual problem may be

solved much more efficiently than the primal problem. The efficiency of the FETI based algorithms for the solution of the frictionless problems was demonstrated by the results of numerical experiments presented by Dostál, Gomes, and Santos [20], Dureisseix and Farhat [27], and most recently by Avery and Farhat [5]. We use the SMALSE-M algorithm presented by Dostal and Kozubek [22] for the solution of elasticity problems with contact boundary conditions using TFETI method Dostal et al [24, 23].

Elasto-plastic processes describe behaviour of solid continuum beyond reversible elastic deformations. They are typically described by hysteresis models with a time memory [12, 47]. The rigorous mathematical analysis of elasto-plastic problems and the numerical methods for their solution started to appear in the late 70ies and in the early 80ies by the work of C. Johnson [40, 41], V. Korneev and U. Langer [44], J. Nečas and I. Hlaváček [50] and others. Since then a lot of mathematical contributions to computational plasticity have been written. Here we refer at least to some of them: to the monographs by J. Simo and T. Hughes [57] and W. Han and B. Reddy [36], to the book of Blaheta [10], to the habilitation theses by C. Carstensen [14] and C. Wieners [65], and to the collection of papers by E. Stein et al. [60].

In this thesis, we consider three different associated elasto-plastic models: the von Mises model with isotropic hardening, the von Mises model with kinematic hardening and the Drucker-Prager perfectly plastic model (see e.g. [36, 10, 51]). The corresponding elasto-plastic constitutive models are discretized by the implicit Euler method in time and consequently a nonlinear stress-strain relation is implemented by the return mapping concept (see e.g. [51, 1, 10]). This approach together with the balance equation, the small strain assumption and a combination of the Dirichlet and Neumann boundary conditions leads to the solution of a nonlinear variational equation with respect to the primal unknown displacement in each time step. Such an equation can also be equivalently formulated as a minimization problem with a potential energy functional (see e.g. [35, 61]).

By a finite element space discretization of the one time step problem, we obtain a system of nonlinear equations. The corresponding nonlinear operator is nondifferentiable but strongly semismooth. Therefore, it is suitable to choose the semismooth Newton method for solving this system since the strong semismoothness together with other properties ensure local quadratic convergence. Semismooth functions in finite dimensional spaces and the semismooth Newton method were introduced in [53]. In elasto-plasticity, the semismoothness was investigated for example in [35, 56, 61, 62].

In each Newton iteration, it is necessary to solve the respective linearized problem. Different linear solvers including those based on multigrid have been successfully tested in [66, 34]. Moreover, since the linear systems of equations corresponding to the elastic and elasto-plastic problems are spectrally equivalent [42], all preconditioners for elastic problems can be applied to elasto-plastic ones as well. As we mentioned, we use the TFETI method to solve the linearized problems.

When we add contact conditions to the classical ones, in each time step of elasto-plasticity problem we obtain a non-linear variational inequality. The non-linearity caused by the elasto-plastic operator can be anihilated by iteration process based on a Newton-

like method. In each iteration we have to solve a problem similar to the one in the case of elasticity with contact boundary conditions. This enables us to use the TFETI method in combination with SMALSE-M algorithm for the solution of the problem.

The outline of the thesis is as follows: In Chapter 1 we give an overview of the elastic problem with classical boundary conditions of Dirichlet and Neumann type. We describe the TFETI method for its efficient parallel solution. Chapter 2 deals with a problem of elasto-plasticity. We present three constitutive models and their time discretization. Further we derive a problem which is solved in each time step in the form of non-linear variational equality. After a spatial discretization the non-linear variational equality is transformed into nonlinear system of algebraic equations solved by non-smooth Newton method. The resulting linearized problems are solved by TFETI method similarly as in the case of elasticity. The elasto-plastic problem is studied in details especially for von Mises model with isotropic hardening. In the case of other models we describe the main differences. In Chapters 3 and 4 we describe only the algebraic formulations of the elasto-plastic problem. In Chapter 3 we generalize the TFETI method for the solution of elastic problem with contact boundary conditions and mention the solver called SMALSE-M. In Chapter 4 we present the algorithm for the solution of elasto-plasticity with contact boundary conditions. This algorithm is based on a suitable combination of algorithms from Chapters 2 and 3. In all chapters we provide numerical examples to, e.g., demonstrate a convergency or scalability of studied algorigthms. Moreover we perform also a comparison of stress states for chosen elastic and elasto-plastic models.

Preliminary

Semismooth functions and semismooth Newton method

Since we want to solve investigated elasto-plastic problems by semismooth Newton method, we start with some notes to semismoothness. For the sake of simplicity, we consider semismoothnes only for finite dimensional spaces. Semismoothness was originally introduced by Mifflin [49] for functionals. Qi and J. Sun [53] extended the definition of semismoothness to vector-valued function to investigate the superlinear convergence of the Newton Method. The semismoothness can be defined in this way.

Definition 1. *Let X, Y be finite dimensional spaces. Let $F : X \to Y$ be locally Lipschitz function in a neighborhood of some $x \in X$. Let $\partial F(x)$ standardly denote the generalized Jacobian in the Clarke sense [16]. F is semismooth at x if*

(i) F is directionally differentiable at x,

(ii) for any $h \in X,\ h \to 0$, and $V \in \partial F(x+h)$,

$$F(x+h) - F(x) - Vh = o(\|h\|). \tag{0.0.1}$$

Moreover, if

$$F(x+h) - F(x) - Vh = O(\|h\|^2) \tag{0.0.2}$$

for any $h \in X,\ h \to 0$, and $V \in \partial F(x+h)$, we say that F is strongly semismooth.

We say that $F :\ X \to Y$ is semismooth (strongly semismooth) on a set $\mathcal{O} \subset X$ if F is semismooth (strongly semismooth) at every point of $\mathcal{O}$.

Convex functions and smooth functions are examples of semismooth (strongly semismooth) functions. Scalar product, sum, compositions of semismooth (strongly semismooth) functions are semismooth (strongly semismooth).

In many applications, it is necessary to solve systems of non-linear equations as follows:

$$F(\boldsymbol{u}) = \boldsymbol{f}, \quad F : \mathbb{R}^n \to \mathbb{R}^n, \quad \boldsymbol{f} \in \mathbb{R}^n.$$

Let us note that in elasto-plasticity, the function F is generally non-differentiable and implicit. If we assume that F is locally Lipschitz continuous, then we can define nonsmooth Newton iterations,

$$\boldsymbol{u}_{j+1} := \boldsymbol{u}_j + \boldsymbol{V}_j^{-1} \left(\boldsymbol{f} - F(\boldsymbol{u}_j) \right), \quad \boldsymbol{V}_j \in \partial F(\boldsymbol{u}_j),$$

where a generalized Jacobians $\boldsymbol{V}_j$ can be taken in a Clarke sense. Local superlinear (quadratic) convergence of the iterates to the solution $\boldsymbol{u}$ is ensured by the following assumptions [53]:

1. $\boldsymbol{u}_0$ is closed to $\boldsymbol{u}$,
2. F is locally Lipschitz function,
3. F is semismooth (strongly semismooth) at $\boldsymbol{u}$,
4. all $\boldsymbol{V} \in \partial F(\boldsymbol{u})$ are non-singular.

From a numerical point of view, it can also be useful to investigate potentiality or strong monotonicity of F. The potentiality implies symmetry of generalized Jacobians and can be useful for a damped Newton method [61]. The strong monotonicity implies that the generalized Jacobians are positive definite and thus the fourth local convergence assumption holds.

Chapter 1

Elasticity without contact

1.1 Classical and variational formulation of elastic problem

Let us consider a deformable body occupying a domain $\Omega \subset \mathbb{R}^3$ with a Lipschitz continuous boundary $\Gamma = \partial\Omega$. We describe the state of the body during a loading process by the Cauchy stress tensor $\sigma \in S$, the displacement $u = (u_1(x), u_2(x), u_3(x)) \in \mathbb{R}^3$ and the small strain tensor $\varepsilon \in S$. Here $S = \mathbb{R}^{3\times 3}_{sym}$ is the space of all symmetric second order tensors.

The above variables depend on the spatial variable $x = (x_1, x_2, x_3) \in \Omega$. The small strain tensor is related to the displacement by the linear relation

$$\varepsilon(u) = \frac{1}{2}\left(\bigtriangledown u + (\bigtriangledown u)^T\right). \tag{1.1.1}$$

We consider the elastic constitutive model given by the Hooke law for isotropic material,

$$\sigma = \mathbb{C}\varepsilon = \lambda \mathrm{tr}(\varepsilon) I + 2\mu\varepsilon \tag{1.1.2}$$

with the Lame coefficients λ, μ. For the sake of simplicity, we assume a homogeneous material, i.e. the constant coefficients $\lambda, \mu > 0$. The trace operator of a tensor is denoted by tr(.) and I denotes the second order identity tensor.

It is useful to introduce the volumetric and deviatoric parts of a tensor $\eta \in S$ by

$$\mathrm{vol}(\eta) = \frac{1}{3}\mathrm{tr}(\eta) I, \quad \mathrm{dev}(\eta) = \eta - \mathrm{vol}(\eta). \tag{1.1.3}$$

It holds that

$$\langle \mathrm{vol}(\eta), \mathrm{dev}(\xi)\rangle_F = 0, \;\; \langle \mathrm{dev}(\eta), \xi\rangle_F = \langle \mathrm{dev}(\eta), \mathrm{dev}(\xi)\rangle_F \quad \forall \eta, \xi \in S. \tag{1.1.4}$$

By (1.1.4), we can find that the fourth order tensor $\mathbb{C}$, defined by (1.1.2), is symmetric and elliptic, i.e.

$$\langle \mathbb{C}\eta, \xi\rangle_F = \langle \eta, \mathbb{C}\xi\rangle_F, \;\; \langle \mathbb{C}\eta, \eta\rangle_F \geq 2\mu\|\eta\|_F^2 \quad \forall \eta, \xi \in S. \tag{1.1.5}$$

The equilibrium equation reads

$$-\operatorname{div}(\sigma(x)) = g(x) \quad \forall x \in \Omega, \tag{1.1.6}$$

where $g(x) = (g_1(x), g_2(x), g_3(x)) \in \mathbb{R}^3$ represents the volume force acting at the point $x \in \Omega$.

Let the boundary Γ be fixed on a part Γ_U that has a nonzero Lebesgue measure with respect to Γ, i.e. we prescribe the homogeneous Dirichlet boundary condition on Γ_U:

$$u(x) = 0 \quad \forall x \in \Gamma_U. \tag{1.1.7}$$

On the rest of the boundary $\Gamma_N = \Gamma \setminus \Gamma_U$, we prescribe the Neumann boundary condition

$$\sigma(x)n(x) = F(x) \quad \forall x \in \Gamma_N, \tag{1.1.8}$$

where $n(x)$ denotes the outward unit normal and $F(x) = (F_1(x), F_2(x), F_3(x))$ denotes a prescribed surface forces at the point $x \in \Gamma_N$. An example of the geometry of Ω with imposed boundary conditions is depicted in Figure 1.1. Similarly, we can consider other boundary conditions, for example symmetry and periodic conditions.

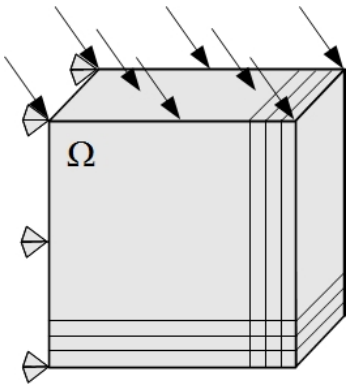

Figure 1.1: An example of the geometry of Ω with imposed boundary conditions.

The classical formulation of the elastic problem is given by (1.1.1), (1.1.2), and (1.1.6)–(1.1.8).

For a weak formulation of the elastic problems, we introduce the space of kinematically admissible displacements,

$$V = \left\{ v \in [H^1(\Omega)]^3 : \ v = 0 \ \text{ on } \ \Gamma_U \right\}. \tag{1.1.9}$$

Then the weak formulation of the elastic problem is following

Problem 1 (Weak formulation of elastic problem)**.**
Find $u = u(x) \in V$ such that

$$a_e(u, v) = \int_\Omega g^T v \, \mathrm{d}x + \int_{\Gamma_N} F^T v \, \mathrm{d}s - \int_\Omega \langle \tau, \varepsilon(v) \rangle_F \, \mathrm{d}x \quad \forall v \in V, \tag{1.1.10}$$

where the bilinear form on V reads

$$a_e(w,v) = \int_\Omega \langle \mathbb{C}\varepsilon(w), \varepsilon(v)\rangle_F \,\mathrm{d}x, \quad w, v \in V. \tag{1.1.11}$$

Here $\varepsilon(v)$ is defined by (1.1.1), $\langle .,. \rangle_F$ and $\|.\|_F$ denote the Frobenius scalar product and the corresponding norm on the space S, respectively. The integral $\int_\Omega \langle \tau, \varepsilon(v)\rangle_F \,\mathrm{d}x$ can represent the nonzero Dirichlet boundary condition or the thermal loading, if we consider it. In next chapter, we will see that the integral represents the linearized stress operator. We assume that the functions σ, F, g, τ are sufficiently smooth to be the integrals in (1.1.10) correctly defined in the Lebesgue sense.

Note that the bilinear form $a_e(w,v)$ is symmetric and V-elliptic on V by (1.1.5) and the Korn inequality [50], i.e.

$$a_e(w,v) = a_e(v,w), \quad \exists c > 0: \quad a_e(v,v) \geq c\|v\|_V^2 \quad \forall v, w \in V. \tag{1.1.12}$$

The mentioned properties of the form a_e ensure that the elastic problem (1.1.10) has a unique solution $u \in V$, for example by Lax-Milgram lemma [50]. Notice that the problem (1.1.10) does not depend on the load history, so it is a static problem.

1.2 Finite element discretization and algebraic formulation

Details to finite element implementation of elasticity can be found for example in [15].

For simplicity, in the theoretical part we assume a polyhedric 3D domain Ω and use the linear simplex elements. The corresponding shape regular triangulation is denoted by $\mathcal{T}_h$. Thus the space V is approximated by its subspace V_h of piecewise linear and continuous functions:

$$V_h := \left\{ v_h \in \left[H^1(\Omega)\right]^3 \;:\; v_h|_T \in [P_1]^3 \quad \forall T \in \mathcal{T}_h,\; v_h|_{\Gamma_U} = 0 \right\}. \tag{1.2.13}$$

In consequence the spaces of the strain and stress states are approximated by piecewise constant functions.

Similarly to (1.1.10) we can formulate the elastic problem after the space discretization. For the sake of simplicity, we do not consider finite element discretization of F, g, and τ.

Problem 2 (Elastic problem after space discretization)**.**
Find $u_h \in V_h$ which solves the variational equation

$$\begin{aligned} &\textstyle\int_\Omega \langle \mathbb{C}(\varepsilon(u_h)), \varepsilon(v_h)\rangle_F \,\mathrm{d}x = \int_\Omega g^T v_h \,\mathrm{d}x + \int_{\Gamma_N} F^T v_h \,\mathrm{d}s \; - \\ &- \textstyle\int_\Omega \langle \tau, \varepsilon(v_h)\rangle_F \,\mathrm{d}x \quad \forall v_h \in V_h. \end{aligned} \tag{1.2.14}$$

We can define the approximated bilinear form a by

$$a(w_h, v_h) = \int_\Omega \langle \mathbb{C}(\varepsilon(w_h)), \varepsilon(v_h) \rangle \, \mathrm{d}x, \quad v_h, w_h \in V_h \tag{1.2.15}$$

and right hand side by

$$b(v_h) = \int_\Omega g^T v_h \, \mathrm{d}x + \int_{\Gamma_N} F^T v_h \, \mathrm{d}s - \int_\Omega \langle \tau, \varepsilon(v_h) \rangle_F \, \mathrm{d}x \quad \forall v_h \in V_h. \tag{1.2.16}$$

Now we can write minimization problem:

$$\text{find } u_h \in V_h : \quad J_h(u_h) \leq J_h(v_h), \quad \forall v_h \in V_h, \tag{1.2.17}$$

where

$$J_h(v_h) = \frac{1}{2} a(v_h, v_h) - b(v_h), \quad v_h \in V_h.$$

Each function $v_h = (v_{h,1}, v_{h,2}, v_{h,3}) \in V_h$ can be represented by a vector

$$\boldsymbol{v} \in \mathbb{R}^n, \ \boldsymbol{v} := (v_{h,j}(x_i))_{i \in \{1,\dots,\mathcal{N}\}, j \in \{1,2,3\}},$$

where $\mathcal{N}$ denotes the number of vertices of the triangulation $\mathcal{T}_h$ and $n = 3\mathcal{N}$ is the number of degrees of freedom. The homogeneous Dirichlet boundary condition is represented by a restriction matrix $\boldsymbol{B}_U \in \mathbb{R}^{m_U \times n}$, i.e.

$$\boldsymbol{B}_U \boldsymbol{u} = \boldsymbol{o}, \tag{1.2.18}$$

with m_U being in number of mesh nodes with imposed Dirichlet boundary condition. Let $\boldsymbol{R}_T \in \mathbb{R}^{12 \times n}$ be a restriction operator for a displacement vector $\boldsymbol{u} \in \mathbb{R}^n$ on a local element $T \in \mathcal{T}_h$, i.e.

$$\boldsymbol{u}_T = \boldsymbol{R}_T \boldsymbol{u}. \tag{1.2.19}$$

We denote the load vector represented the forces f, g, τ by $\boldsymbol{f}$.

Further, we use a vector representation in $\mathbb{R}^6$ of the stress and strain tensors that is typical for an implementation of elastic problem, i.e.

$$\boldsymbol{\sigma} = (\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{13})^T, \quad \boldsymbol{\varepsilon} = (\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, 2\varepsilon_{12}, 2\varepsilon_{23}, 2\varepsilon_{13})^T. \tag{1.2.20}$$

Notice that the stress and strain vectors have different structures in comparison to the above tensor notation. Let $\boldsymbol{\sigma}_T$ be the algebraic representation of $\sigma_h = \mathbb{C}(\varepsilon(u_h))$ on an element $T \in \mathcal{T}_h$.

Now, we introduce the algebraic representation $\boldsymbol{C} \in \mathbb{R}^{6 \times 6}$ of the Hooke tensor $\mathbb{C}$. The form of the matrix $\boldsymbol{C}$ is

$$\boldsymbol{C} := \begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix}. \tag{1.2.21}$$

We also introduce the matrix $\boldsymbol{G} \in \mathbb{R}^{6\times 12}$ representing the algebraical relation between the strain and the displacement (the exact form of $\boldsymbol{G}$ is in [2]), i.e. the strain $\boldsymbol{\varepsilon_T}$ on an element $T \in \mathcal{T}_h$ can be found by (1.2.19) in the form

$$\boldsymbol{\varepsilon}_T = \boldsymbol{G}\boldsymbol{R}_T\boldsymbol{u}. \tag{1.2.22}$$

Based on the introduced notation and a local to global summing convention we define the stiffness matrix $\boldsymbol{K} \in \mathbb{R}^{n\times n}$

$$\boldsymbol{K} = \sum_{T\in\mathcal{T}_h} \left(\boldsymbol{C}\boldsymbol{G}\boldsymbol{R}_T\right)^T \boldsymbol{G}\boldsymbol{R}_T, \tag{1.2.23}$$

which represents the bilinear form a_h.

Let

$$\boldsymbol{V} := \left\{\boldsymbol{v} \in \mathbb{R}^n \mid \boldsymbol{B}_U\boldsymbol{v} = \boldsymbol{o}\right\}. \tag{1.2.24}$$

Then by using (1.2.23), we can rewrite the equation (1.2.14) as follows: find $\boldsymbol{u} \in \boldsymbol{V}$ such that

$$\boldsymbol{v}^T\left(\boldsymbol{K}\boldsymbol{u} - \boldsymbol{f}\right) = 0 \quad \forall \boldsymbol{v} \in \boldsymbol{V}. \tag{1.2.25}$$

Let $\tilde{\boldsymbol{u}} \in \mathbb{R}^{n-m_U}$, $\tilde{\boldsymbol{f}} \in \mathbb{R}^{n-m_U}$, and $\tilde{\boldsymbol{K}} \in \mathbb{R}^{(n-m_U)\times(n-m_U)}$ denote the restrictions of $\boldsymbol{u}$, $\boldsymbol{f}$, and $\boldsymbol{K}$ with respect to the Dirichlet conditions. Then we can rewrite the equation (1.2.25) to the following system of linear equations:

$$\text{find } \boldsymbol{u} \in \boldsymbol{V}: \quad \tilde{\boldsymbol{K}}\tilde{\boldsymbol{u}} = \tilde{\boldsymbol{f}}. \tag{1.2.26}$$

Let us note that $\boldsymbol{K}$ is a symmetric and positive semidefinite matrix and $\tilde{\boldsymbol{K}}$ is a symmetric and positive definite matrix. Therefore the linearized problem (1.2.26) has a unique solution. The problem (1.2.26) can be equivalently rewritten as a minimization problem:

$$\text{find } \boldsymbol{u} \in \boldsymbol{V}: \quad \boldsymbol{J}(\boldsymbol{u}) \le \boldsymbol{J}(\boldsymbol{v}),\ \forall \boldsymbol{v} \in \boldsymbol{V}, \tag{1.2.27}$$

where

$$\boldsymbol{J}(\boldsymbol{v}) = \frac{1}{2}\boldsymbol{v}^T\boldsymbol{K}\boldsymbol{v} - \boldsymbol{f}^T\boldsymbol{v}, \quad \boldsymbol{v} \in \boldsymbol{V}.$$

1.3 Total-FETI domain decomposition method

To apply the TFETI domain decomposition to the minimization problem (1.2.17), we tear the body from the part of the boundary with the Dirichlet boundary condition, decompose it into subdomains, assign to each subdomain a unique number, and introduce new "gluing" conditions on the artificial intersubdomain boundaries and on the boundaries with imposed Dirichlet condition (see Figure 1.2).

In particular, the polyhedric domain Ω is decomposed into a system of s disjoint polyhedric subdomains $\Omega^p \subset \mathbb{R}^3$, $p = 1, 2, \ldots, s$. We assume that the decomposition is consistent with the triangulation $\mathcal{T}_h$, i.e.

$$\forall T \in \mathcal{T}_h\ \exists!\ p \in \{1, 2, \ldots, s\}: \quad T \subset \bar{\Omega}^p \tag{1.3.28}$$

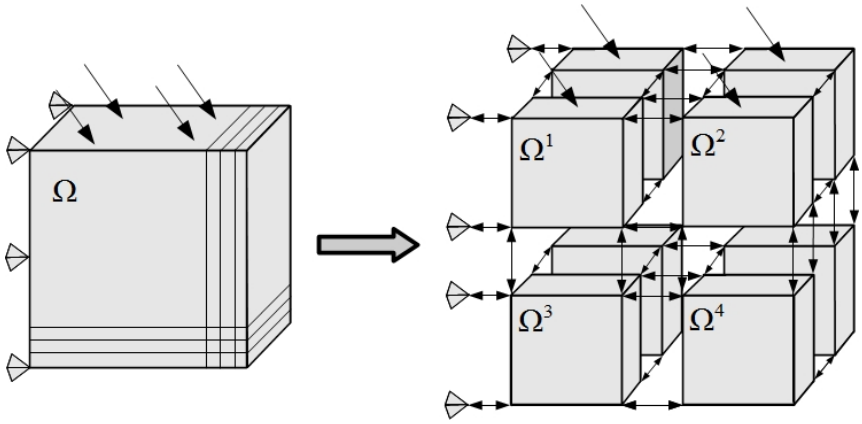

Figure 1.2: TFETI domain decomposition with subdomain renumbering

and define

$$\mathcal{T}_h^p := \{T \in \mathcal{T}_h : \ T \subset \bar{\Omega}^p\}, \quad \mathcal{T}_h = \bigcup_{p\in\{1,2,\ldots,s\}} \mathcal{T}_h^p. \tag{1.3.29}$$

After the decomposition each boundary Γ^p of Ω^p consists of three disjoint parts Γ_U^p, Γ_N^p, and Γ_G^p, $\Gamma^p = \overline{\Gamma}_U^p \cup \overline{\Gamma}_N^p \cup \overline{\Gamma}_G^p$, where

$$\Gamma_U^p = \Gamma_U \cap \Gamma^p, \ \ \Gamma_N^p = \Gamma_N \cap \Gamma^p, \ \ \Gamma_G^p = \bigcup_{q\in\{1,2,\ldots,s\}\setminus\{p\}} \Gamma_G^{pq},$$

with Γ_G^{pq} being the part of Γ^p which is glued to Ω^q, $p \neq q$.

Similarly to the definition of V_h in (1.2.13), we can define the spaces V_h^p, $p = 1, 2, \ldots s$, of piecewise linear and continuous approximations of $[H^1(\Omega^p)]^3$:

$$V_h^p := \left\{ v_h^p \in \left[H^1(\Omega^p)\right]^3 : \ v_h^p|_T \in [P_1]^3 \ \ \forall T \in \mathcal{T}_h^p, \ \ v_h^p|_{\Gamma_U^p} = 0 \right\}. \tag{1.3.30}$$

Let $\mathtt{V}_h := V_h^1 \times V_h^2 \times \ldots \times V_h^s$ be a product space and

$$\mathtt{K}_h := \{\mathtt{v}_h = (v_h^1, \ldots, v_h^s) \in \mathtt{V}_h : \ v_h^p = v_h^q \ \text{ on } \Gamma_G^{pq} \ \ \forall p, q \in \{1, 2, \ldots, s\}, \ p \neq q\}. \tag{1.3.31}$$

Let us note that we slightly distinguish the notation introduced for the TFETI method from the notation introduced in Sections 1.2. For example, we write $\mathtt{V}_h$ for the TFETI method while in Section 1.2, we used the notation V_h. A similar distinction is also introduced for the below algebraic description of the TFETI method.

Let

$$\begin{aligned} \mathtt{J}_h(\mathtt{v}_h) \ \ = \ \ & \sum_{p=1}^{s} \left\{ \frac{1}{2} \int_{\Omega^p} \langle \mathbb{C}\varepsilon(v_h^p), \varepsilon(v_h^p) \rangle_F dx - \int_{\Omega^p} g^T v_h^p \, \mathrm{d}x - \right. \\ & \left. - \int_{\Gamma_N^p} F^T v_h^p \, \mathrm{d}s + \int_{\Omega^p} \langle \tau, \varepsilon(v_h^p) \rangle_F \, \mathrm{d}x \right\} \end{aligned} \tag{1.3.32}$$

be a functional defined on $\mathtt{V}_h$. Then the minimization problem (1.2.17) can be equivalently rewritten into the form:

$$\text{find } \mathtt{u}_h \in \mathtt{K}_h : \quad \mathtt{J}_h(\mathtt{u}_h) \leq \mathtt{J}_h(\mathtt{v}_h) \quad \forall \mathtt{v}_h \in \mathtt{K}_h, \tag{1.3.33}$$

where $\mathtt{u}_h = ((u_h)|_{\Omega^1}, \ldots, (u_h)|_{\Omega^s})$ and $u_h \in V_h$ solves (1.2.17).

Each function $\mathtt{v}_h = (v_h^1, v_h^2, \ldots, v_h^s)$, $\mathtt{v}_h \in \mathtt{V}_h$, can be represented by a vector $\mathbf{v} \in \mathbb{R}^{\mathtt{n}}$, $\mathbf{v} = \left(\boldsymbol{v}_1^T, \boldsymbol{v}_2^T, \ldots, \boldsymbol{v}_s^T\right)^T$, where $\boldsymbol{v}_p \in \mathbb{R}^{n_p}, p \in \{1, 2, \ldots, s\}$, is the algebraic representation of v_h^p and $\mathtt{n} = \sum_{p=1}^{s} n_p$. Similarly we can assemble the vector $\mathbf{f} \in \mathbb{R}^{\mathtt{n}}$, $\mathbf{f} = \left(\boldsymbol{f}_1^T, \boldsymbol{f}_2^T, \ldots, \boldsymbol{f}_s^T\right)^T$, $\boldsymbol{f}_p \in \mathbb{R}^{n_p}, p \in \{1, 2, \ldots, s\}$, such that $\boldsymbol{f}_p$ is the algebraic representation of the load restricted to Ω^p and Γ_N^p. Let the matrix $\mathbf{B}_G \in \mathbb{R}^{\mathtt{m}_G \times \mathtt{n}}$ represent the gluing conditions introduced in (1.3.31) and $\mathbf{B}_U \in \mathbb{R}^{\mathtt{m}_U \times \mathtt{n}}$ the Dirichlet boundary conditions introduced in (1.3.30). More details about matrices $\mathbf{B}_G$ and $\mathbf{B}_U$ can be found in [46]. Both matrices can be combined into one constraint matrix

$$\mathbf{B}_E = \begin{bmatrix} \mathbf{B}_G \\ \mathbf{B}_U \end{bmatrix}, \quad \mathbf{B}_E \in \mathbb{R}^{\mathtt{m} \times \mathtt{n}}, \quad \mathtt{m} = \mathtt{m}_G + \mathtt{m}_U.$$

Typically $\mathtt{m}$ is much smaller than $\mathtt{n}$. Let the matrix $\mathbf{K} \in \mathbb{R}^{\mathtt{n} \times \mathtt{n}}$, $\mathbf{K} = diag(\boldsymbol{K}_1, \boldsymbol{K}_2, \ldots, \boldsymbol{K}_s)$ denote a symmetric positive semidefinite block diagonal matrix, where

$$\boldsymbol{K}_p = \sum_{T \in \mathcal{T}_h^p} |T| \left(\boldsymbol{C}\boldsymbol{G}\boldsymbol{R}_T^p\right)^T \boldsymbol{G}\boldsymbol{R}_T^p, \quad \boldsymbol{K}_p \in \mathbb{R}^{n_p \times n_p}.$$

Here $\boldsymbol{R}_T^p \in \mathbb{R}^{12 \times n_p}$ is a restriction operator for a displacement vector $\boldsymbol{u}_p \in \mathbb{R}^{n_p}$ to a local element $T \in \mathcal{T}_h^p$. The diagonal blocks $\boldsymbol{K}_p$, $p \in \{1, 2, \ldots, s\}$, which correspond to the subdomains Ω^p, are positive semidefinite sparse matrices with known kernels spanned by the respective rigid body modes.

The algebraical formulation of (1.3.33) is following:

$$\text{find } \mathbf{u} \in \mathbf{V} \ : \ \mathbf{J}(\mathbf{u}) \leq \mathbf{J}(\mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{V}, \tag{1.3.34}$$

where

$$\mathbf{J}(\mathbf{v}) := \frac{1}{2}\mathbf{v}^T\mathbf{K}\mathbf{v} - \mathbf{f}^T\mathbf{v}$$

and

$$\mathbf{V} := \{\mathbf{v} \in \mathbb{R}^{\mathtt{n}} : \ \mathbf{B}_E\mathbf{v} = \mathbf{o}\}.$$

Even though (1.3.34) is a standard convex quadratic programming problem, its formulation is not suitable for numerical solution. The reasons are that $\mathbf{K}$ is typically ill-conditioned, singular, and very large.

The complications mentioned above may be essentially reduced by applying the duality theory of convex programming (see, e.g., Dostál [19]), where all the constraints are enforced by the Lagrange multipliers $\boldsymbol{\lambda}$. The Lagrangian associated with the problem (1.3.34) is

$$L(\mathbf{v}, \boldsymbol{\lambda}) = \mathbf{J}(\mathbf{v}) + \boldsymbol{\lambda}^T\mathbf{B}_E\mathbf{v}. \tag{1.3.35}$$

It is well known [19] that (1.3.34) is equivalent to the saddle point problem:

$$\text{find } (\mathbf{u},\boldsymbol{\lambda}) \in \mathbb{R}^{\mathtt{n}} \times \mathbb{R}^{\mathtt{m}} : \quad L(\mathbf{u},\boldsymbol{\nu}) \leq L(\mathbf{u},\boldsymbol{\lambda}) \leq L(\mathbf{v},\boldsymbol{\lambda}) \quad \forall (\mathbf{v},\boldsymbol{\nu}) \in \mathbb{R}^{\mathtt{n}} \times \mathbb{R}^{\mathtt{m}} \tag{1.3.36}$$

in sense that $\mathbf{u}$ solves (1.3.34) if and only if $(\mathbf{u},\boldsymbol{\lambda})$ solves (1.3.36).

1.3.1 Optimal solvers to equality constrained problems

The solution of (1.3.36) leads to the equivalent problem to find $(\mathbf{u},\boldsymbol{\lambda}) \in \mathbb{R}^{\mathtt{n}} \times \mathbb{R}^{\mathtt{m}}$ satisfying:

$$\begin{pmatrix} \mathbf{K} & \mathbf{B}_E^T \\ \mathbf{B}_E & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{o} \end{pmatrix}. \tag{1.3.37}$$

The system (1.3.37) is uniquely solvable which is guaranteed by the following necessary and sufficient conditions [9]:

$$Ker\,\mathbf{B}_E^T = \mathbf{o}, \tag{1.3.38}$$

$$Ker\,\mathbf{K} \cap Ker\,\mathbf{B}_E = \mathbf{o}. \tag{1.3.39}$$

Notice that (1.3.38) is the condition on the full row-rank of $\mathbf{B}_E$. Let us mention that an orthonormal basis of $Ker\,\mathbf{K}$ is known à-priori and that its vectors are columns of $\mathbf{R} \in \mathbb{R}^{\mathtt{n}\times l}$, $l = \mathtt{n} - rank(\mathbf{K})$.

The first equation in (1.3.37) is satisfied if

$$\mathbf{f} - \mathbf{B}_E^T\boldsymbol{\lambda} \in Im\,\mathbf{K}. \tag{1.3.40}$$

Then we can express $\mathbf{u}$ in the form:

$$\mathbf{u} = \mathbf{K}^{\dagger}(\mathbf{f} - \mathbf{B}_E^T\boldsymbol{\lambda}) + \mathbf{R}\boldsymbol{\alpha} \tag{1.3.41}$$

for an appropriate $\boldsymbol{\alpha} \in \mathbb{R}^l$ and arbitrary matrix $\mathbf{K}^{\dagger}$ satisfying $\mathbf{K}\mathbf{K}^{\dagger}\mathbf{K} = \mathbf{K}$. Here $\mathbf{K}^{\dagger}$ is a generalized inverse matrix whose application on a vector can be efficiently implemented, see Remark 1.3.1.

Remark 1.3.1. *The diagonal block $\boldsymbol{K}_p$, $p \in \{1,2,\ldots,s\}$, which corresponds to the subdomain Ω^p, is positive semidefinite sparse matrix with known kernel basis created by the rigid body modes. This is a great advantage because all blocks can be effectively regularized without extra fill in and then decomposed using any standard sparse Cholesky type factorization method for nonsingular matrices [13]. We completely avoid problems with zero pivots. The action of $\mathbf{K}^{\dagger}$ on a vector is naturally parallelized with respect to the subdomains and computed using backward and forward substitutions.*

Remark 1.3.2. *Notice that the choice of $\mathbf{K}^{\dagger}$ is invariant in the exact arithmetic but not in the computer one. The computation may be stabilized by choosing the Moore-Penrose inverse whose action can be carried out for the same computational cost using orthogonal projector onto $Im\,\mathbf{K}$ (see [48]).*

The condition (1.3.40) can be equivalently written as

$$\mathbf{R}^T(\mathbf{f} - \mathbf{B}_E^T\boldsymbol{\lambda}) = \mathbf{o}. \tag{1.3.42}$$

Further substituting (1.3.41) into the second equation in (1.3.37) we arrive at

$$-\mathbf{B}_E\mathbf{K}^\dagger\mathbf{B}_E^T\boldsymbol{\lambda} + \mathbf{B}_E\mathbf{R}\boldsymbol{\alpha} = -\mathbf{B}_E\mathbf{K}^\dagger\mathbf{f}. \tag{1.3.43}$$

Summarizing (1.3.43) and (1.3.42) we find that the pair $(\boldsymbol{\lambda}, \boldsymbol{\alpha}) \in \mathbb{R}^{\mathtt{m}} \times \mathbb{R}^l$ satisfies:

$$\begin{pmatrix} \mathbf{F} & \mathbf{N}^T \\ \mathbf{N} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \begin{pmatrix} \mathbf{d} \\ \mathbf{e} \end{pmatrix}, \tag{1.3.44}$$

where $\mathbf{F} := \mathbf{B}_E\mathbf{K}^\dagger\mathbf{B}_E^T$, $\mathbf{N} := -\mathbf{R}^T\mathbf{B}_E^T$, $\mathbf{d} := \mathbf{B}_E\mathbf{K}^\dagger\mathbf{f}$, and $\mathbf{e} := -\mathbf{R}^T\mathbf{f}$.

Since $\mathbf{N}$ is of full row-rank as follow from (1.3.39), the inverse $(\mathbf{N}\mathbf{N}^T)^{-1}$ exists and $\mathbf{P}_\mathbf{N} := \mathbf{I} - \mathbf{N}^T(\mathbf{N}\mathbf{N}^T)^{-1}\mathbf{N}$ is well defined and represents the orthogonal projector onto $Ker\,\mathbf{N}$. Applying $\mathbf{P}_\mathbf{N}$ to the first equation in (1.3.44) and checking that $\mathbf{P}_\mathbf{N}\mathbf{N}^T\boldsymbol{\alpha} = \mathbf{o}$ we eliminate $\boldsymbol{\alpha}$ and obtain that $\boldsymbol{\lambda}$ satisfies:

$$\mathbf{P}_\mathbf{N}\mathbf{F}\boldsymbol{\lambda} = \mathbf{P}_\mathbf{N}\mathbf{d}, \quad \mathbf{N}\boldsymbol{\lambda} = \mathbf{e}. \tag{1.3.45}$$

In practical computations, we further decompose $\boldsymbol{\lambda} = \boldsymbol{\lambda}_{Im} + \boldsymbol{\lambda}_{Ker}$ into two orthogonal components $\boldsymbol{\lambda}_{Im} \in Im\,\mathbf{N}^T$ and $\boldsymbol{\lambda}_{Ker} \in Ker\,\mathbf{N}$, substitute them into (1.3.45) and get the problem

$$\mathbf{P}_\mathbf{N}\mathbf{F}\boldsymbol{\lambda}_{Ker} = \mathbf{P}_\mathbf{N}\left(\mathbf{d} - \mathbf{F}\boldsymbol{\lambda}_{Im}\right) \text{ on } Ker\,\mathbf{N}, \tag{1.3.46}$$

with $\boldsymbol{\lambda}_{Im} = \mathbf{N}^T\left(\mathbf{N}\mathbf{N}^T\right)^{-1}\mathbf{e}$. Equation (1.3.46) is solved efficiently by the projected conjugate gradient method with preconditioning (PCGP) [29] using the lumped preconditioner to $\mathbf{F}$ in the form $\overline{\mathbf{F}^{-1}} = \mathbf{B}_E\mathbf{K}\mathbf{B}_E^T$. For more information see [29]. We obtain the following algorithmic scheme for the solution of (1.3.34):

Algorithm 1 (Linear solver based on the TFETI method)**.**

Set $\mathbf{N} := -\mathbf{R}^T\mathbf{B}_E^T$, $\mathbf{H} := (\mathbf{N}\mathbf{N}^T)^{-1}$, $\mathbf{d} := \mathbf{B}_E\mathbf{K}^\dagger\mathbf{f}$, *and* $\mathbf{e} := -\mathbf{R}^T\mathbf{f}$.
Compute $\boldsymbol{\lambda}_{Im} := \mathbf{N}^T\mathbf{H}\mathbf{e}$.
Set $\tilde{\mathbf{d}} := \mathbf{d} - \mathbf{F}\boldsymbol{\lambda}_{Im}$.
Compute $\boldsymbol{\lambda}_{Ker}$ *from* (1.3.46) *by PCGP:*
  $\mathbf{r}^0 = \tilde{\mathbf{d}}$, $\boldsymbol{\lambda}_{Ker}^0 = \mathbf{o}$.
  for $j = 1, 2, \ldots,$ *until convergence* **do**
    Project $\mathbf{w}^{j-1} = \mathbf{P}_\mathbf{N}\mathbf{r}^{j-1}$.
    Precondition $\mathbf{z}^{j-1} = \overline{\mathbf{F}^{-1}}\mathbf{w}^{j-1}$.
    Re-project $\mathbf{y}^{j-1} = \mathbf{P}_\mathbf{N}\mathbf{z}^{j-1}$.
    $\boldsymbol{\beta}^j = (\mathbf{y}^{j-1})^T\mathbf{w}^{j-1}/(\mathbf{y}^{j-2})^T\mathbf{w}^{j-2}$ $\quad (\boldsymbol{\beta}^1 = 0)$.
    $\mathbf{p}^j = \mathbf{y}^{j-1} + \boldsymbol{\beta}^j\mathbf{p}^{j-1}$ $\quad (\mathbf{p}^1 = \mathbf{y}^0)$.

12: $\boldsymbol{\gamma}^j = (\mathbf{y}^{j-1})^T \mathbf{w}^{j-1} / (\mathbf{p}^j)^T \mathbf{F} \mathbf{p}^j$.
13: $\boldsymbol{\lambda}_{Ker}^j = \boldsymbol{\lambda}_{Ker}^{j-1} + \boldsymbol{\gamma}^j \mathbf{p}^j$.
14: $\mathbf{r}^j = \mathbf{r}^{j-1} - \boldsymbol{\gamma}^j \mathbf{F} \mathbf{p}^j$.
15: **if** $\|\mathbf{w}^{j-1}\| \leq \epsilon_{PCGP} \|r^0\|$ **then stop.**
16: **end for**
17: $\boldsymbol{\lambda}_{Ker} = \boldsymbol{\lambda}_{Ker}^j$.
18: *Set* $\boldsymbol{\lambda} := \boldsymbol{\lambda}_{Im} + \boldsymbol{\lambda}_{Ker}$.
19: *Compute* $\boldsymbol{\alpha} := \mathbf{HN}(\mathbf{d} - \mathbf{F}\boldsymbol{\lambda})$.
20: *Compute* $\mathbf{u} := \mathbf{K}^{\dagger}(\mathbf{f} - \mathbf{B}_E^T \boldsymbol{\lambda}) + \mathbf{R}\boldsymbol{\alpha}$.

Remark 1.3.3. *Action of* $\mathbf{H}$ *on a vector may be efficiently implemented by the sparse Cholesky factorization of* $\mathbf{NN}^T$.

1.4 Comments to plane strain problem

In our work we also consider a dimensional reduction of 3D problem based on plane strain. In this section we briefly describe the assumptions for plane strain and the main differences between 3D and plane strain problems. More details can be found in [50, 51].

We consider an elastic body represented by a vertical cylinder. Its cross section is a general domain $\Omega' \subset \mathbb{R}^2$ and on its ends zeroth normal displacement are prescribed. The scheme is depicted in Figure 1.3.

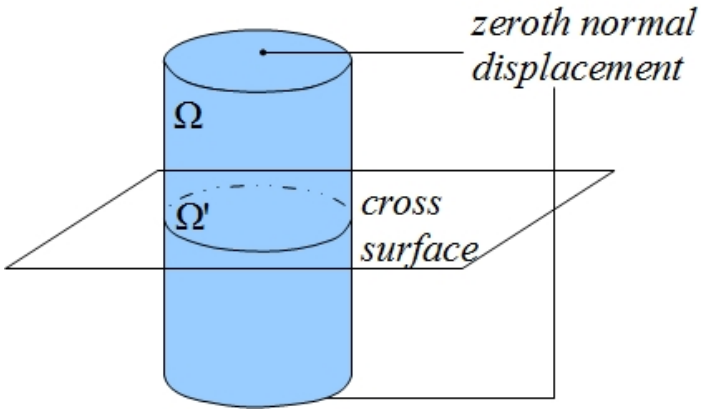

Figure 1.3: Vertical cylinder Ω with cross section Ω'

Next, we suppose that the material coefficients, body forces, and boundary conditions on the curved surface of the cylinder do not depend on the coordinate x_3 provided that the x_3-axis is parallel to the generators of the cylinder. Finally, we suppose that the third components $g_3(x) = F_3(x) = 0$ of g and F, respectively.

Under these assumptions we can put

$$u_3 = 0, \quad \frac{\partial u_1}{\partial x_3} = \frac{\partial u_2}{\partial x_3} = 0 \tag{1.4.47}$$

and thus sufficient representation of the displacement vector u at $x = (x_1, x_2)$ in Ω is following: $u(x) = (u_1(x_1, x_2), u_2(x_1, x_2))^T$.

From (1.4.47) we obtain that $\varepsilon_{i3} = \varepsilon_{3i} = 0$ for $i \in \{1, 2, 3\}$. Based on the Hooke law, we have the following schemes of the strain and stress tensors

$$\varepsilon = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & 0 \\ \varepsilon_{12} & \varepsilon_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & 0 \\ \sigma_{12} & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix}, \text{ where } \sigma_{33} = \frac{\lambda}{2(\mu + \lambda)}(\sigma_{11} + \sigma_{22}). \tag{1.4.48}$$

When we substitute the strain tensor ε into (1.1.10), we obtain the equation with the same structure but defined only in two dimensional domain.

Thus the variable σ_{33} is not necessary for displacement computation in elasticity, but this variable is important for computing of nonlinear operator in the next chapter.

Also the structure of the algebraic formulation and the TFETI method remain the same. We have only different sizes of the considered vectors and matrices. For example a function $v_h = (v_{h,1}, v_{h,2})$ is represented by

$$\boldsymbol{v} \in \mathbb{R}^n, \ \boldsymbol{v} := (v_{h,j}(x_i))_{i \in \{1,\ldots,\mathcal{N}\}, j \in \{1,2\}},$$

the strain and stress vector representations are

$$\boldsymbol{\varepsilon} = (\varepsilon_{11}, \varepsilon_{22}, 2\varepsilon_{12})^T, \quad \boldsymbol{\sigma} = (\sigma_{11}, \sigma_{22}, \sigma_{12})^T, \tag{1.4.49}$$

and the form of the matrix $\boldsymbol{C}$ is

$$\boldsymbol{C} := \begin{bmatrix} \lambda + 2\mu & \lambda & 0 \\ \lambda & \lambda + 2\mu & 0 \\ 0 & 0 & \mu \end{bmatrix}. \tag{1.4.50}$$

1.5 Numerical experiments

In this section, we illustrate the functionality of the introduced TFETI solver on two benchmarks considering linear elastic material model. Elasto-plastic material model and scalability analysis will be considered in the next chapter. The above algorithms were implemented in `MatSol` library [45] developed in parallel Matlab environment. All computations are performed using maximum 28 cores with 2GB memory per core of the HP Blade system, model BLc7000. The stopping tolerance of the PCGP algorithm is

$$\epsilon_{PCGP} = 10^{-7}. \tag{1.5.51}$$

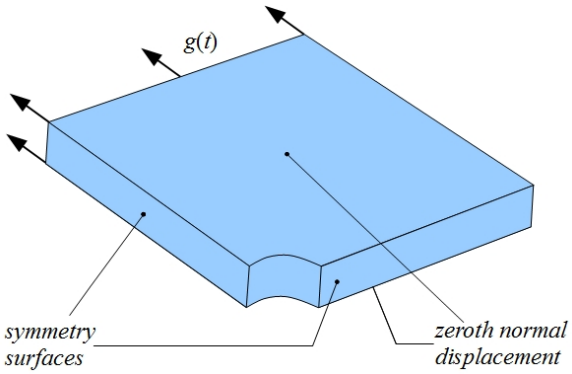

Figure 1.4: Geometry of Ω

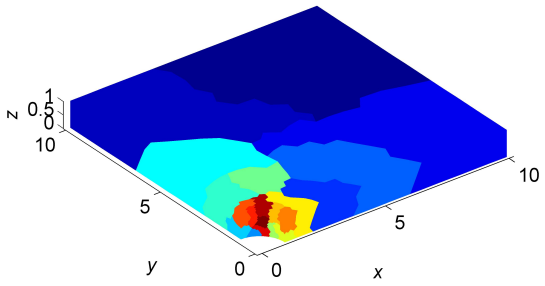

Figure 1.5: Domain decomposition into 20 subdomains

1.5.1 The beam with a hole

Let us consider infinite beam with a hole in the center. To compute this problem, we assume only a part of the beam, with thickness of the body and with the corresponding symmetry conditions (zero normal displacement) depicted in Figure 1.4.

The body Ω is made of homegeneous isotropic material with the parameters $E = 206\,900$, $\nu = 0.29$, where E and ν are the Young modulus and the Poisson ratio, respectively, which are related with the Lamé coefficients λ and μ by the following formulas:

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} = 110\,743.8, \quad \mu = \frac{E}{2(1+\nu)} = 801\,938.$$

The indicated traction forces are prescribed by the function $g = g(t) = 450$.

The mesh is generated in Ansys and has 4 250 nodes and 19 008 tetrahedrons. For illustrative purposes, we decompose the body Ω into 20 subdomains by Metis, and depict it in Figure 1.5. After decomposition we have 16 467 primal variables and 5 314 dual variables.

The problem was solved by Algorithm 1 implemented in MatSol. Distributions of the von Mises stress $\|\mathrm{dev}(\sigma)\|_F$ and the total displacement $\|u\|$ are depicted in Figures 1.6 and 1.7, respectively.

1.5.2 Äspö Pillar Stability Experiment

The investigated benchmark is related to the Äspö Pillar Stability Experiment (APSE) [3, 4, 11]. The APSE was carried out to examine the failure process in a heterogeneous and slightly fractured granite rock mass. The pillar arises between two 1.75m diameter boreholes drilled from the floor of the access tunnel, which is excavated for the experiment purposes in the depth of 450m under surface. One of the main objectives of the experiment was to estimate the yield strength of the rock mass.

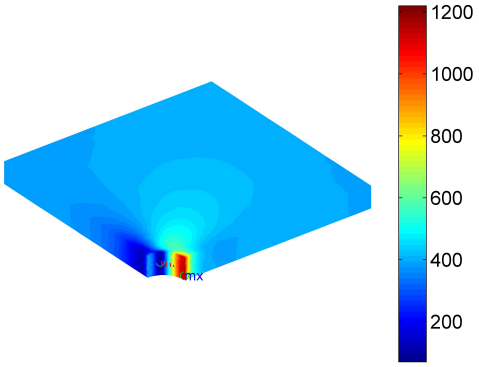

Figure 1.6: Distribution of von Mises stress

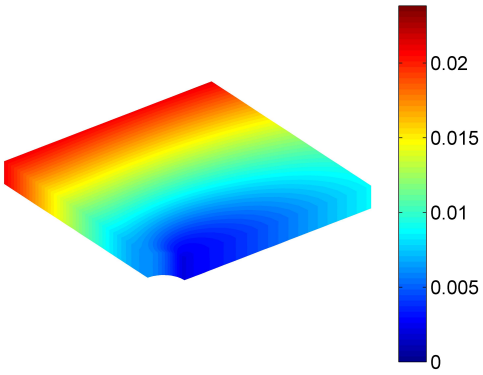

Figure 1.7: Total displacement

We start with a simplified plane strain modelling of the pillar failure. The corresponding two dimensional rectangular geometry represents a horizontal cross-section of the pillar 2m below the access tunnel. The geometry is depicted in Figure 1.8, where the size of Ω is 27.42×31.42 and the distance between two 1.75 diameter boreholes (at the narrowest part of the pillar) is 1m. We assume only one quarter of the geometry, due to symmetry of the problem. The generated mesh has 5 316 nodes and 9 696 triangles. We decompose Ω into 8 subdomains by Metis, this is depicted in Figure 1.9. After decomposition we have 10 632 primal variables and 817 dual variables.

We use the following material parameters: $E = 55 \cdot 10^9$, $\nu = 0.25$. The indicated pressures are prescribed by the functions $g_1(t) = 13 \cdot 10^6$ and $g_2(t) = 46 \cdot 10^6$.

The problem is solved by Algorithm 1. In Figure 1.10, we depict the total displacement. In the remaining figures we zoom the situation near the hole, because this is the place, which is very interesting for us. Particularly, we show distribution of von Mises stress $\|\mathrm{dev}(\sigma)\|_F$, normal stresses σ_{xx}, σ_{yy}, and shear stress σ_{xy} after loading in Figures 1.11, 1.12, 1.13, and 1.14, respectively.

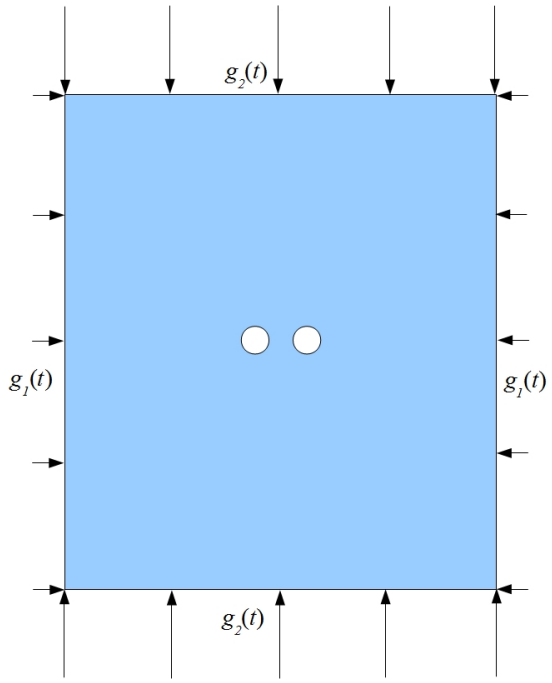

Figure 1.8: Geometry of Ω

Figure 1.9: Domain decomposition into 8 subdomains

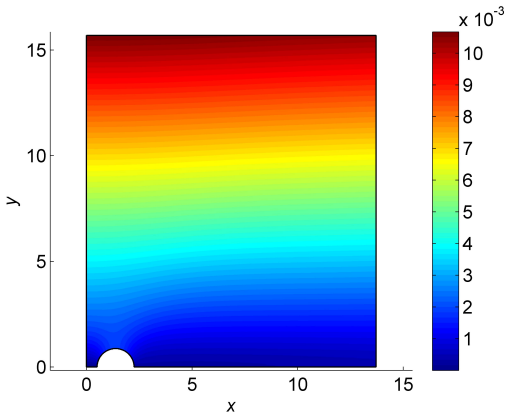

Figure 1.10: Total displacement

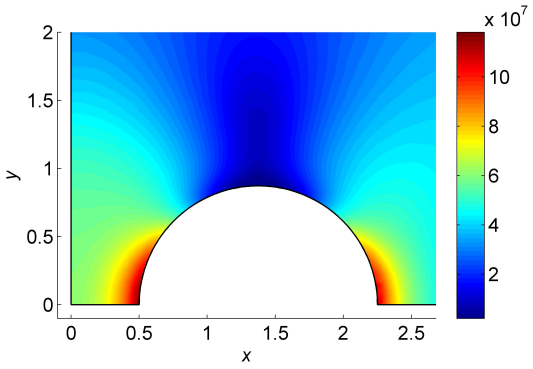

Figure 1.11: Distribution of von Mises stress

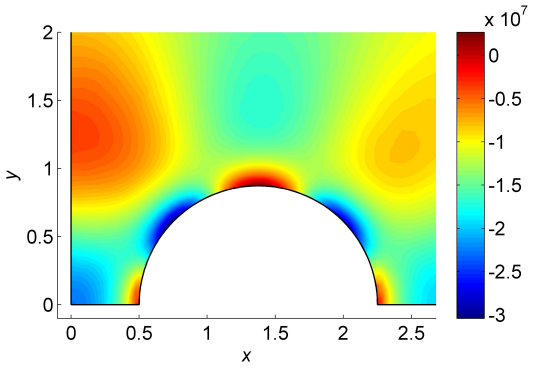

Figure 1.12: Distribution of stress σ_{xx}

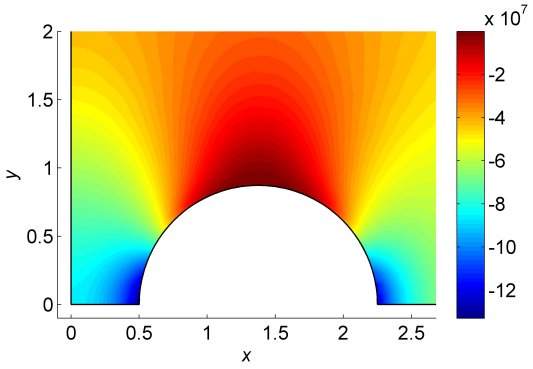

Figure 1.13: Distribution of stress σ_{yy}

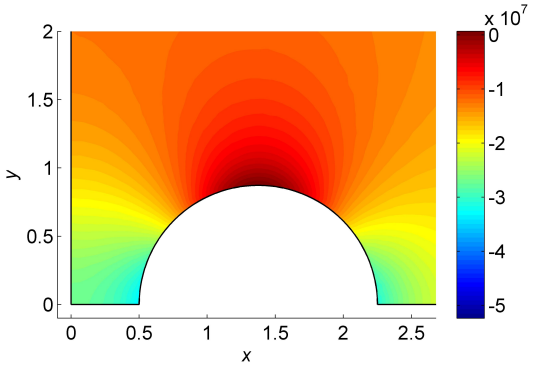

Figure 1.14: Distribution of stress σ_{xy}

Chapter 2

Elasto-plasticity without contact

In contrast to elasticity, the elasto-plastic models are non-linear from a certain stress state. In the following, we consider time dependent constitutive models in which so-called internal variables representing hardening parameters may be included. The internal variables can also depend on the time variable $t \in Q = [t_0, t^*]$.

The chapter is organized as follows: In Sections 2.1 and 2.2 we introduce the scheme of elasto-plastic constitutive model and its time discretization on example of the von Mises criterion with linear isotropic hardening. In Section 2.3 we summarize the von Mises model with kinematic hardening and perfect plastic Drucker-Prager model. In Section 2.4 we formulate a time discretized elasto-plastic problem in form of non-linear variational equation. For the sake of simplicity we confine only to the von Mises model with isotropic hardening. In Section 2.5 we consider a spatial discerization of the problem and introduce the corresponding algebraic formulation in the form of a system of non-linear equations. This system is solved by the semismooth Newton method. The corresponding linearized problems we solve by TFETI domain decomposition method, analogously as in elastic problem, see Section 2.6. In Section 2.8 we present numerical experiments for considered elasto-plastic models, summarize convergence results and scalability experiments.

In this chapter we follow our article [17] and the book [10]. From the engineering point of view, the elasto-plastic problem is described in detail for example in [51], from mathematical point of view in [36].

2.1 Elasto-plastic initial value constitutive model

In comparison to elasticity, elasto-plasticity is time dependent model where the history of loading is taken into account. We assume associated elasto-plasticity with von Mises plastic criterion and linear isotropic hardening law (see e.g. [36, 10, 51]). The elasto-plastic initial-value constitutive model consists of the following components:

1. Additive decomposition of the strain tensor into the elastic and plastic parts:

$$\varepsilon = \varepsilon^e + \varepsilon^p. \tag{2.1.1}$$

2. Linear elastic law between the stress and the elastic strain:

$$\sigma = \mathbb{C}\varepsilon^e, \tag{2.1.2}$$

where the fourth order tensor $\mathbb{C}$ is defined by (1.1.2).

3. The von Mises yield function coupled with an isotropic hardening variable κ:

$$\Phi(\sigma, \kappa) = \sqrt{\frac{3}{2}}\|\mathrm{dev}(\sigma)\|_F - (\sigma_y + H_m\kappa) \leq 0, \tag{2.1.3}$$

where $\sigma_y, H_m > 0$ denote the initial yield stress and the hardening modulus, respectively.

4. The associated plastic flow rule:

$$\dot{\varepsilon}^p = \dot{\gamma}\frac{\partial\Phi}{\partial\sigma} = \dot{\gamma}\sqrt{\frac{3}{2}}\frac{\mathrm{dev}(\sigma)}{\|\mathrm{dev}(\sigma)\|_F}, \quad \dot{\gamma} \geq 0, \tag{2.1.4}$$

where $\dot{\varepsilon}^p$ and $\dot{\gamma}$ denote the time derivative of the plastic strain and the plastic multiplier, respectively.

5. The hardening law based on the accumulated plastic strain rate:

$$\dot{\kappa} = \sqrt{\frac{2}{3}}\|\dot{\varepsilon}^p\|_F = \dot{\gamma}. \tag{2.1.5}$$

Notice that the second equality in (2.1.5) follows from (2.1.4).

6. The loading/unloading conditions:

$$\dot{\gamma} \geq 0, \;\; \Phi(\sigma, \kappa) \leq 0, \;\; \dot{\gamma}\Phi(\sigma, \kappa) = 0. \tag{2.1.6}$$

7. The initial conditions:

$$\varepsilon(x, t_0) = \varepsilon^e(x, t_0) = \varepsilon^p(x, t_0) = \sigma(x, t_0) = 0, \;\; \kappa(x, t_0) = 0, \quad x \in \Omega. \tag{2.1.7}$$

The weak formulation of the corresponding elasto-plastic problem can be found in [36]. Here we only consider a time discretized elasto-plastic model.

2.2 Time discretized elasto-plastic constitutive model

Let us consider the following discretization of the time interval

$$t_0 < t_1 < \ldots < t_k < \ldots < t_N = t^*.$$

Let us denote $\sigma_k = \sigma_k(x) = \sigma(x, t_k)$, $x \in \Omega$ and similarly for other variables. To approximate the time derivatives, we use the implicit Euler method. However, it is also possible to use for example the Crank-Nicholson scheme. Then by (2.1.1) and (2.1.2) we get

$$\dot{\varepsilon}^p(t_{k+1}) \approx \frac{\varepsilon^p_{k+1} - \varepsilon^p_k}{\triangle t_{k+1}} = \frac{\mathbb{C}^{-1}(\sigma^{trial}_{k+1} - \sigma_{k+1})}{\triangle t_{k+1}}, \quad \triangle t_{k+1} = t_{k+1} - t_k, \tag{2.2.8}$$

where

$$\sigma_{k+1}^{trial} = \sigma_k + \mathbb{C}\triangle\varepsilon_{k+1}, \quad \triangle\varepsilon_{k+1} = \varepsilon_{k+1} - \varepsilon_k. \tag{2.2.9}$$

By (2.1.1)–(2.2.9), we can formulate the time discretized elasto-plastic constitutive problem as follows. Given the values σ_k, κ_k, ε_k of the stress, the isotropic hardening, and the strain, respectively, at the time t_k and given the incremental strain $\triangle\varepsilon_{k+1}$ for the interval $[t_k, t_{k+1}]$, solve the following system of algebraic equations

$$\mathbb{C}^{-1}(\sigma_{k+1}^{trial} - \sigma_{k+1}) = \triangle\gamma_{k+1}\sqrt{\frac{3}{2}}\frac{\mathrm{dev}(\sigma_{k+1})}{\|\mathrm{dev}(\sigma_{k+1})\|_F} \tag{2.2.10}$$

$$\kappa_{k+1} - \kappa_k = \triangle\gamma_{k+1} \tag{2.2.11}$$

for the unknowns σ_{k+1}, κ_{k+1}, and $\triangle\gamma_{k+1}$, subject to the constraints

$$\triangle\gamma_{k+1} \geq 0, \;\; \Phi(\sigma_{k+1}, \kappa_{k+1}) \leq 0, \;\; \triangle\gamma_{k+1}\Phi(\sigma_{k+1}, \kappa_{k+1}) = 0. \tag{2.2.12}$$

This constitutive problem can be solved explicitly by the return mapping concept (see e.g. [10, 51]). It means that we firstly apply the elastic predictor, i.e. we verify whether $\Phi(\sigma_{k+1}^{trial}, \kappa_k) \leq 0$. If it holds then

$$\triangle\gamma_{k+1} = 0, \quad \sigma_{k+1} = \sigma_{k+1}^{trial}. \tag{2.2.13}$$

If $\Phi(\sigma_{k+1}^{trial}, \kappa_k) > 0$, then by the plastic corrector we have

$$\triangle\gamma_{k+1} = \frac{1}{3\mu + H_m}\Phi(\sigma_{k+1}^{trial}, \kappa_k), \quad \sigma_{k+1} = \sigma_{k+1}^{trial} - \frac{3\mu}{3\mu + H_m}\sqrt{\frac{2}{3}}\hat{n}(\sigma_{k+1}^{trial}), \tag{2.2.14}$$

where

$$\hat{n}(\tau) = \frac{\mathrm{dev}(\tau)}{\|\mathrm{dev}(\tau)\|_F}, \quad \tau \in S. \tag{2.2.15}$$

Notice that the second formula in (2.2.14) is correctly defined since the denominator $\|\mathrm{dev}(\sigma_{k+1}^{trial})\|_F > 0$ for $\Phi(\sigma_{k+1}^{trial}, \kappa_k) > 0$. Let us define the stress and hardening operators $T_\sigma(\tau, \omega; .) : S \to S$, $T_\kappa(\tau, \omega; .) : S \to S$ with respect to parameters $\tau \in S$, $\omega \in \mathbb{R}_+$, such that for $\xi \in S$

$$T_\sigma(\tau, \omega; \xi) := \mathbb{C}\xi - \frac{3\mu}{3\mu + H_m}\sqrt{\frac{2}{3}}\Phi^+(\tau + \mathbb{C}\xi, \omega)\hat{n}(\tau + \mathbb{C}\xi), \tag{2.2.16}$$

$$T_\kappa(\tau, \omega; \xi) := \frac{1}{3\mu + H_m}\Phi^+(\tau + \mathbb{C}\xi, \omega), \tag{2.2.17}$$

respectively, where Φ^+ denotes the positive part of the function Φ. Then by (2.2.9), (2.2.11), (2.2.13), (2.2.14), (2.2.16) and (2.2.17),

$$\triangle\kappa_{k+1} = T_\kappa(\sigma_k, \kappa_k; \triangle\varepsilon_{k+1}), \quad \triangle\sigma_{k+1} = T_\sigma(\sigma_k, \kappa_k; \triangle\varepsilon_{k+1}). \tag{2.2.18}$$

For the sake of brevity, we denote the stress operator $T_\sigma(\sigma_k, \kappa_k; .)$ with respect to the current parameters σ_k and κ_k by $T_k(.)$. By [10, 35, 61], the operator $T_k : S \to S$ is potential, Lipschitz continuous, strongly monotone, and strongly semismooth on S.

Here we choose the Clark generalized derivative T_k^o of T_k in the following way:

1. If $\Phi(\sigma_k + \mathbb{C}\xi, \kappa_k) \leq 0$, then
$$T_k^o(\xi) = \mathbb{C}. \tag{2.2.19}$$

2. If $\Phi(\sigma_k + \mathbb{C}\xi, \kappa_k) > 0$, then

$$\begin{aligned} T_k^o(\xi) &= \mathbb{C} - \frac{3\mu}{3\mu + H_m}\sqrt{\frac{2}{3}}\frac{\partial\Phi(\sigma_k + \mathbb{C}\xi, \kappa_k)}{\partial\xi}\hat{n}(\sigma_k + \mathbb{C}\xi) - \\ &\quad - \frac{3\mu}{3\mu + H_m}\sqrt{\frac{2}{3}}\Phi(\sigma_k + \mathbb{C}\xi, \kappa_k)\frac{\partial\hat{n}(\sigma_k + \mathbb{C}\xi)}{\partial\xi}, \end{aligned} \tag{2.2.20}$$

where

$$\begin{aligned} \frac{\partial\Phi(\sigma_k + \mathbb{C}\xi, \kappa_k)}{\partial\xi} &= 2\mu\sqrt{\frac{3}{2}}\hat{n}(\sigma_k + \mathbb{C}\xi), \\ \frac{\partial\hat{n}(\sigma_k + \mathbb{C}\xi)}{\partial\xi} &= 2\mu\frac{\mathbb{I}_d - \hat{n}(\sigma_k + \mathbb{C}\xi) \otimes \hat{n}(\sigma_k + \mathbb{C}\xi)}{\|\mathrm{dev}(\sigma_k + \mathbb{C}\xi)\|_F}, \\ \mathbb{I}_d\xi &:= \mathrm{dev}(\xi), \quad \forall\xi \in S. \end{aligned}$$

Here the symbol $\otimes$ denotes tenzor product. Notice that T_k is not differentiable at $\xi \in S$, $\Phi(\sigma_k + \mathbb{C}\xi, \kappa_k) = 0$. Otherwise $T_k^o(\xi) = \partial T_k(\xi)/\partial\xi$.

Let us recall that the stress, strain, hardening, and displacement variables also depend on a spatial variable $x \in \Omega$. We consider the dependence of $T_k(\triangle\varepsilon_k)$ on x in the following sense:

$$T_k(\triangle\varepsilon_k) = T_k(\triangle\varepsilon_k)(x) := T_\sigma(\sigma_k(x), \kappa_k(x); \triangle\varepsilon_k(x)). \tag{2.2.21}$$

2.3 Other investigated elasto-plastic models

In previous sections we introduced the von Mises model with isotropic hardening. In this section we show, that the other elasto-plastic models have a similar scheme. Concretely, we assume the von Mises model with kinematic hardening and perfect plastic Drucker-Prager model.

2.3.1 The von Mises model with kinematic hardening

Compared to the previous model with the scalar variable κ for isotropic hardening, now we assume the model with a tensor variable β representing kinematic hardening.

The yield function Φ is for this case given by

$$\Phi(\sigma, \beta) = \sqrt{\frac{3}{2}}\|\mathrm{dev}(\sigma - \beta)\|_F - \sigma_y. \tag{2.3.22}$$

The associated plastic flow rule and the hardening law have the forms

$$\begin{aligned}\dot{\varepsilon}^p &= \dot{\gamma}\frac{\partial\Phi}{\partial\sigma} = \dot{\gamma}\sqrt{\tfrac{3}{2}}\,\hat{n}(\sigma-\beta),\\ c_0^{-1}\dot{\beta} &= -\dot{\gamma}\frac{\partial\Phi}{\partial\beta} = \dot{\gamma}\sqrt{\tfrac{3}{2}}\,\hat{n}(\sigma-\beta),\end{aligned} \tag{2.3.23}$$

where $\hat{n}$ is defined by (2.2.15) and $c_0 > 0$ is a material constant.

In the next text we will use variables and relations after time discretization. The trial stress σ_{k+1}^{trial} and increment of strain $\Delta\varepsilon_{k+1}$ are same as in (2.2.9).

The analogical equations to (2.2.10) and (2.2.11) have the following forms:

$$\mathbb{C}^{-1}(\sigma_{k+1}^{trial} - \sigma_{k+1}) = \triangle\gamma_{k+1}\sqrt{\frac{3}{2}}\,\hat{n}(\sigma_{k+1}-\beta_{k+1}) \tag{2.3.24}$$

$$c_0^{-1}\,(\beta_{k+1}-\beta_k) = \triangle\gamma_{k+1}\sqrt{\frac{3}{2}}\,\hat{n}(\sigma_{k+1}-\beta_{k+1}) \tag{2.3.25}$$

for the unknown σ_{k+1}, β_{k+1}, and $\triangle\gamma_{k+1}$, subject to the constraints

$$\triangle\gamma_{k+1} \geq 0,\ \ \Phi(\sigma_{k+1},\beta_{k+1}) \leq 0,\ \ \triangle\gamma_{k+1}\Phi(\sigma_{k+1},\beta_{k+1}) = 0. \tag{2.3.26}$$

This constitutive problem can also be solved explicitly by the return mapping concept.

Let us define the stress and hardening operators $T_\sigma(\tau,\omega;.) : S \to S$, $T_\beta(\tau,\omega;.) : S \to S$ with respect to parameters $\tau \in S$, $\omega \in S$, such that for $\xi \in S$

$$T_\sigma(\tau,\omega;\xi) := \mathbb{C}\xi - \frac{2\mu}{2\mu+c_0}\sqrt{\frac{2}{3}}\,\Phi^+(\tau+\mathbb{C}\xi,\omega)\,\hat{n}(\tau+\mathbb{C}\xi-\omega), \tag{2.3.27}$$

$$T_\beta(\tau,\omega;\xi) := \sqrt{\frac{2}{3}}\frac{c_0}{2\mu+c_0}\Phi^+(\tau+\mathbb{C}\xi,\omega)\,\hat{n}(\tau+\mathbb{C}\xi-\omega), \tag{2.3.28}$$

respectively, where Φ^+ denotes the positive part of the function Φ. Then by (2.2.9), (2.3.25), (2.3.27), and (2.3.28) we get

$$\triangle\beta_{k+1} = T_\beta(\sigma_k,\beta_k;\triangle\varepsilon_{k+1}), \quad \triangle\sigma_{k+1} = T_\sigma(\sigma_k,\beta_k;\triangle\varepsilon_{k+1}). \tag{2.3.29}$$

The incremental stress-strain operator $T_k(\xi) := T_\sigma(\sigma_k,\beta_k;\xi)$, $\xi \in S$ is potential, Lipschitz continuous, strongly monotone and strongly semismooth on S, similarly to the previous model.

We choose the Clark generalized derivative T_k^o of T_k in the following way:

1. If $\Phi(\sigma_k+\mathbb{C}\xi,\beta_k) \leq 0$, then
$$T_k^o(\xi) = \mathbb{C}. \tag{2.3.30}$$
2. If $\Phi(\sigma_k+\mathbb{C}\xi,\beta_k) > 0$, then $T_k^o(\xi) = \dfrac{\partial T_k(\xi)}{\partial\xi}$, i.e.

$$\begin{aligned}T_k^o(\xi) = {} & \mathbb{C} - 2\mu\tfrac{2\mu}{2\mu+c_0}\mathbb{I}_d - \\ & -2\mu\tfrac{2\mu}{2\mu+c_0}\sqrt{\tfrac{2}{3}}\tfrac{\sigma_y}{\|\mathrm{dev}(\mathbb{C}\xi-\beta_k)\|_F}\left(\hat{n}(\mathbb{C}\xi-\beta_k)\otimes\hat{n}(\mathbb{C}\xi-\beta_k) - \mathbb{I}_d\right).\end{aligned} \tag{2.3.31}$$

2.3.2 The perfect plastic model with the Drucker-Prager criterion

In this model we do not use any hardening variable. The yield function is for this case given by

$$\Phi(\sigma) := \sqrt{\frac{1}{2}}\|\mathrm{dev}(\sigma)\|_F + \eta p(\sigma) - \overline{c}, \tag{2.3.32}$$

where $\eta,\ \overline{c} > 0$ are the material parameters and

$$p(\sigma) = \frac{1}{3}\mathrm{tr}(\sigma).$$

The associated plastic flow rule has the form

$$\dot{\varepsilon}^p = \dot{\gamma}\frac{\partial \Phi}{\partial \sigma} = \dot{\gamma}\left(\frac{\sqrt{2}}{2}\,\hat{n}(\sigma) + \frac{\eta}{3}I\right), \tag{2.3.33}$$

where $\hat{n}$ is defined by (2.2.15).

As in the previous model we consider variables and relations after time discretization. The variables σ_{k+1}^{trial} and $\Delta\varepsilon_{k+1}$ are the same as in relation (2.2.9).

The analogical equation to (2.2.10) has the following form:

$$\mathbb{C}^{-1}(\sigma_{k+1}^{trial} - \sigma_{k+1}) = \triangle\gamma_{k+1}\left(\frac{\sqrt{2}}{2}\hat{n}(\sigma_{k+1}) \ + \frac{\eta}{3}I\right) \tag{2.3.34}$$

for the unknowns σ_{k+1} and $\triangle\gamma_{k+1}$, subject to the constraints

$$\triangle\gamma_{k+1} \geq 0, \ \ \Phi(\sigma_{k+1}) \leq 0, \ \ \triangle\gamma_{k+1}\Phi(\sigma_{k+1}) = 0. \tag{2.3.35}$$

This constitutive problem can be also solved explicitly by the return mapping scheme, see e.g. [10, 51]. Let us define the stress operator $T_\sigma(\tau;.) : S \to S$ with respect to a parameter $\tau \in S$ such that for $\xi \in S$

$$\begin{aligned} T_\sigma(\tau;\xi) \ := \ & \left(\|\mathrm{dev}(\tau + \mathbb{C}\xi)\|_F - \tfrac{3\mu\sqrt{2}\Phi^+(\tau+\mathbb{C}\xi)}{3\mu+\eta^2(3\lambda+2\mu)}\right)^+ \hat{n}(\tau+\mathbb{C}\xi) + \\ & + \left[\tfrac{\overline{c}}{\eta} - \left(-p(\tau+\mathbb{C}\xi) + \tfrac{\eta(3\lambda+2\mu)\Phi^+(\tau+\mathbb{C}\xi)}{3\mu+\eta^2(3\lambda+2\mu)} + \tfrac{\overline{c}}{\eta}\right)^+\right] I - \tau, \end{aligned} \tag{2.3.36}$$

where Φ^+ denotes the positive part of the function Φ. Notice that $T_\sigma(\tau;\cdot)$ can also be continuously defined at $\xi \in S$ in which $\mathrm{dev}(\tau + \mathbb{C}\xi) = 0$. Then by (2.2.9) and (2.3.36),

$$\triangle\sigma_{k+1} = T_\sigma(\sigma_k; \triangle\varepsilon_{k+1}). \tag{2.3.37}$$

The incremental stress-strain operator $T_k(\xi) := T_\sigma(\sigma_k;\xi),\ \xi \in S$ by [62] is potential, Lipschitz continuous, monotone and strongly semismooth on S, similarly to the previous models, but not strongly monotone since we only consider perfect plasticity.

Now we can choose the Clark generalized derivative T_k^o of T_σ in the following way:

1. If $\xi \in S$: $\Phi(\sigma_k + \mathbb{C}\xi) < 0$ then

$$T_k(\xi) = \mathbb{C}\xi$$

and

$$T_k^o(\xi) = \frac{\partial T_k(\xi)}{\partial \xi} = \mathbb{C}. \tag{2.3.38}$$

2. If $\xi \in S$: $\Phi(\sigma_k + \mathbb{C}\xi) > 0 \wedge \|\mathrm{dev}(\sigma_k + \mathbb{C}\xi)\| > \dfrac{3\mu\sqrt{2}\Phi(\sigma_k + \mathbb{C}\xi)}{3\mu + \eta^2(3\lambda + 2\mu)}$ then

$$T_k(\xi) = \mathbb{C}\xi - \frac{3\Phi(\sigma_k + \mathbb{C}\xi)}{3\mu + \eta^2(3\lambda + 2\mu)} \left\{ \mu\sqrt{2}\, \hat{n}(\sigma_k + \mathbb{C}\xi) + \frac{\eta}{3}(3\lambda + 2\mu) I \right\}$$

and

$$\begin{aligned} T_k^o(\xi) \;=\; & \frac{\partial T_k(\xi)}{\partial \xi} = \mathbb{C} - \frac{3}{3\mu + \eta^2(3\lambda + 2\mu)} \left\{ \frac{\partial \Phi(\sigma_k + \mathbb{C}\xi)}{\partial \xi} \otimes \frac{\partial \Phi(\sigma_k + \mathbb{C}\xi)}{\partial \xi} + \right. \\ & \left. + \frac{2\mu^2\sqrt{2}\Phi(\sigma_k + \mathbb{C}\xi)}{\|\mathrm{dev}(\sigma_k + \mathbb{C}\xi)\|} \left(\mathbb{I}_d - \hat{n}(\sigma_k + \mathbb{C}\xi) \otimes \hat{n}(\sigma_k + \mathbb{C}\xi) \right) \right\}, \end{aligned} \tag{2.3.39}$$

where

$$\frac{\partial \Phi(\sigma_k + \mathbb{C}\xi)}{\partial \xi} = \mu\sqrt{2}\, \hat{n}(\sigma_k + \mathbb{C}\xi) + \frac{\eta}{3}(3\lambda + 2\mu) I.$$

3. If $\xi \in S$: $\Phi(\sigma_k + \mathbb{C}\xi) > 0 \wedge \|\mathrm{dev}(\sigma_k + \mathbb{C}\xi)\| < \dfrac{3\mu\sqrt{2}\Phi(\sigma_k + \mathbb{C}\xi)}{3\mu + \eta^2(3\lambda + 2\mu)}$ then

$$T_k(\xi) = \frac{\overline{c}}{\eta} I - \sigma_k$$

and

$$T_k^o(\xi) = \frac{\partial T_k(\xi)}{\partial \xi} = \mathbb{O} \quad \text{zeroth tensor of fourth order} \tag{2.3.40}$$

For $\xi \in S$, $\Phi(\sigma_k + \mathbb{C}\xi) = 0$, we set $T_k^o(\xi) = \mathbb{C}$, for $\xi \in S$, $\Phi(\sigma_k + \mathbb{C}\xi) > 0$ and $\|\mathrm{dev}(\sigma_k + \mathbb{C}\xi)\| = \dfrac{3\mu\sqrt{2}\Phi(\sigma_k + \mathbb{C}\xi)}{3\mu + \eta^2(3\lambda + 2\mu)}$, we set $T_k^o(\xi) = \mathbb{O}$.

2.4 Elasto-plastic problem in incremental form

Within this section we again confine only to the von Mises model with isotropic hardening, similarly we can operate with other models.

Let us introduce the elasto-plastic problem in an incremental form which is more convenient from the computational point of view. Simply we substitute the stress operator T_k, defined by (2.2.16) into the incremental form of the balance equation

$$\int_\Omega \langle \triangle\sigma_{k+1}, \varepsilon(v)\rangle_F \,\mathrm{d}x = \int_\Omega \triangle g_{k+1}^T v \,\mathrm{d}x + \int_{\Gamma_F} \triangle F_{k+1}^T v \,\mathrm{d}s \quad \forall v \in V, \forall t \in [t_0, t^*], \quad (2.4.41)$$

with loading increment $\triangle F_{k+1} = F_{k+1} - F_k$, $\triangle g_{k+1} = g_{k+1} - g_k$ to obtain the time discretized elasto-plastic problem in the following incremental form.

Problem 3 (Elasto-plastic problem in the incremental form)**.**
Given the stress state field $\sigma_k \in [L^2(\Omega)]^{3\times 3}_{sym}$ and the state of the isotropic hardening field $\kappa_k \in L^2(\Omega)$ at the time t_k, find the displacement $u_{k+1} = u_k + \triangle u_{k+1} \in V$, where the increment $\triangle u_{k+1} \in V$ solves the variational equation

$$\int_\Omega \langle T_k(\varepsilon(\triangle u_{k+1})), \varepsilon(v)\rangle_F \,\mathrm{d}x = \int_\Omega \triangle g_{k+1}^T v \,\mathrm{d}x + \int_{\Gamma_N} \triangle F_{k+1}^T v \,\mathrm{d}s \quad \forall v \in V, \quad (2.4.42)$$

Set the stress and isotropic hardening fields $\sigma_{k+1} = \sigma_k + \triangle\sigma_{k+1}$, $\kappa_{k+1} = \kappa_k + \triangle\kappa_{k+1}$ in the next time step t_{k+1} from relations

$$\triangle\kappa_{k+1} = T_\kappa(\sigma_k, \kappa_k; \varepsilon(\triangle u_{k+1})), \quad \triangle\sigma_{k+1} = T_\sigma(\sigma_k, \kappa_k; \varepsilon(\triangle u_{k+1})), \quad (2.4.43)$$

almost everywhere in Ω.

Problem 3 can be equivalently formulated as a minimization problem [35, 61]. Since the operator T_k is strongly monotone and Lipschitz continuous on S, the non-linear equation (2.4.42) has a unique solution $\triangle u_{k+1} \in V$ (see e.g. [32]). Let us note that the choice of the space V is not fully correct for perfect plasticity since the operator T_k is not strongly monotone on V in such a case, see e.g. [7, 59]. Nevertheless also perfect plasticity problem are usually solved with respect to unknown displacement for reasonable loads.

As we will see in the next section, we have to solve a linearized problem in each Newton iteration. To do this, it is useful to define the bilinear form $a_k(u) : V \times V \to \mathbb{R}$ for $u \in V$ by

$$a_k(u)(w, v) = \int_\Omega \langle T_k^o(\varepsilon(u))\varepsilon(w), \varepsilon(v)\rangle \,\mathrm{d}x, \quad v, w \in V, \quad (2.4.44)$$

where the operator $T_k^o(.) = T_k^o(.)(x)$ is defined by (2.2.19) and (2.2.20). Since the operator T_k is potential, Lipschitz continuous and strongly monotone on S, the bilinear form a_k is symmetric and V-elliptic on V.

2.5 Finite element discretization and algebraic formulation

Details to finite element discretization of elasto-plastic problems can be found in [15] and [10, 35].

Similarly, as in Chapter 1, we assume a polyhedric 3D domain Ω and use the linear simplex elements. The corresponding shape regular triangulation is denoted by $\mathcal{T}_h$ and V_h is subspace of piecewise linear and continuous functions constructed over $\mathcal{T}_h$. Therefore the spaces of the strains, the stress and the isotropic hardening are approximated by piecewise constant functions.

Similarly to (2.4.42), (2.4.43) we can formulate the one time-step elasto-plastic problem after the space discretization. Let $\sigma_{k,h}$, $\kappa_{k,h}$ be piecewise constant stress and hardening variables with respect to the triangulation $\mathcal{T}_h$ at time t_k obtained from a previous time process.

Problem 4 (One time step of elasto-plasticity)**.**
Find the displacement $u_{k+1,h} = u_{k,h} + \triangle u_{k+1,h} \in V_h$, where the increment $\triangle u_{k+1,h} \in V_h$ solves the variational equation

$$\int_\Omega \langle T_{k,h}(\varepsilon(\triangle u_{k+1,h})), \varepsilon(v_h)\rangle_F \,\mathrm{d}x = \int_\Omega \triangle g_{k+1}^T v_h \,\mathrm{d}x + \int_{\Gamma_N} \triangle F_{k+1}^T v_h \,\mathrm{d}s \quad \forall v_h \in V_h, \tag{2.5.45}$$

where $T_{k,h}(.) := T_\sigma(\sigma_{k,h}, \kappa_{k,h}; .)$. Set the stress and isotropic hardening fields $\sigma_{k+1,h} = \sigma_{k,h} + \triangle\sigma_{k+1,h}$, $\kappa_{k+1,h} = \kappa_{k,h} + \triangle\kappa_{k+1,h}$ in the next time step t_{k+1} from the relations

$$\triangle\sigma_{k+1,h} = T_\sigma(\sigma_{k,h}, \kappa_{k,h}; \varepsilon(\triangle u_{k+1,h})), \quad \triangle\kappa_{k+1,h} = T_\kappa(\sigma_{k,h}, \kappa_{k,h}; \varepsilon(\triangle u_{k+1,h})) \tag{2.5.46}$$

for every elements of $\mathcal{T}_h$.

For the sake of simplicity, we do not consider any aproximation of F_k and g_k. Similarly as in (2.2.19) and (2.2.20), we can define the generalized derivative $T_{k,h}^o$ of $T_{k,h}$ and consequently also define the approximated bilinear form $a_{k,h}(u_h)$ for $u_h \in V_h$ by

$$a_{k,h}(u_h)(w_h, v_h) = \int_\Omega \langle T_{k,h}^o(\varepsilon(u_h))\varepsilon(w_h), \varepsilon(v_h)\rangle \,\mathrm{d}x, \quad v_h, w_h \in V_h. \tag{2.5.47}$$

Also we use the matrices $\boldsymbol{B}_U$, $\boldsymbol{R}_T$, $\boldsymbol{C}$, $\boldsymbol{G}$ and vectors $\boldsymbol{\sigma}$, $\boldsymbol{\varepsilon}$ introduced in Chapter 1. Because we consider time discretized problem, we denote by $\boldsymbol{u}_k$ and $\triangle\boldsymbol{u}_{k+1}$ the displacement vector and the computed displacement increment at the time step k, respectively. We denote the load vector represented the volume and surface forces f_k, g_k by $\boldsymbol{f}_k$ and its increment by $\triangle\boldsymbol{f}_k$.

Recall that the stress and strain vectors have different structures in comparison to the above tensor notation, see Section 1.2. Therefore we must carefully distinguish this difference in algebraic representation of the operators T_σ, T_κ, $T_{k,h}$, and $T_{k,h}^o$. The vectors in sense of stress variables are denoted by letters $\boldsymbol{\sigma}$, $\boldsymbol{\tau}$, the vectors in sense of strain variables are denoted by letters $\boldsymbol{\varepsilon}$, $\boldsymbol{\varepsilon}^p$, $\boldsymbol{\eta}$, and $\boldsymbol{\xi}$. Let $\boldsymbol{\sigma}_{k,T}$ and $\boldsymbol{\kappa}_{k,T}$ be the algebraic representations of $\sigma_{k,h}$ and $\kappa_{k,h}$ on an element $T \in \mathcal{T}_h$, respectively.

We introduce the algebraic representations $\boldsymbol{E}_\varepsilon \in \mathbb{R}^{6\times6}$, $\boldsymbol{E}_\sigma \in \mathbb{R}^{6\times6}$, $\|.\|_\sigma$, $\boldsymbol{\Phi}$, $\hat{\boldsymbol{n}}$, $\boldsymbol{T}_{\kappa,k,T}$, $\boldsymbol{T}_{k,T}$, and $\boldsymbol{T}_{k,T}^o$ of the deviatoric operator $\mathbb{I}_d$ related to the strain and stress variables, the Frobenius norm with respect to a stress variable, the functions Φ, $\hat{n}$, and the restrictions

of the functions $T_\kappa\left(\sigma_k|_T, \kappa_k|_T, \cdot\right)$, $T_{k,h}$, $T^o_{k,h}$ on $T \in \mathcal{T}_h$, respectively, with respect to the vector form (1.2.20) of the stress and strain variables. The forms of matrices $\boldsymbol{E}_\varepsilon$, $\boldsymbol{E}_\sigma$, and the norm $\|.\|_\sigma$ are

$$\boldsymbol{E}_\varepsilon := \frac{1}{3}\begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.5 \end{bmatrix},$$

$$\boldsymbol{E}_\sigma := \boldsymbol{P}\boldsymbol{E}_\varepsilon, \quad \boldsymbol{P} := diag(1,1,1,2,2,2),$$

and

$$\|\boldsymbol{\tau}\|_\sigma := \left(\boldsymbol{\tau}^T \boldsymbol{P}\boldsymbol{\tau}\right)^{1/2}, \quad \boldsymbol{\tau} \in \mathbb{R}^6,$$

respectively. Consequently the functions $\boldsymbol{\Phi}$, $\hat{\boldsymbol{n}}$, $\boldsymbol{T}_{\kappa,k,T}$, $\boldsymbol{T}_{k,T}$, $\boldsymbol{T}^o_{k,T}$ are defined by (2.1.3), (2.2.15), (2.2.17), (2.2.16), (2.2.19), (2.2.20) in the following way:

$$\boldsymbol{\Phi}(\boldsymbol{\tau}, \boldsymbol{\kappa}) := \sqrt{\frac{3}{2}}\|\boldsymbol{E}_\sigma \boldsymbol{\tau}\|_\sigma - \left(\sigma_y + H_m \boldsymbol{\kappa}\right),$$

$$\hat{\boldsymbol{n}}(\boldsymbol{\tau}) := \frac{\boldsymbol{E}_\sigma(\boldsymbol{\tau})}{\|\boldsymbol{E}_\sigma(\boldsymbol{\tau})\|_\sigma}, \quad \overline{\boldsymbol{n}}_{k,T}(\boldsymbol{\xi}) = \hat{\boldsymbol{n}}\left(\boldsymbol{\sigma}_{k,T} + \boldsymbol{C}\boldsymbol{\xi}\right),$$

$$\boldsymbol{T}_{\kappa,k,T}(\boldsymbol{\xi}) := \frac{1}{3\mu + H_m}\boldsymbol{\Phi}^+\left(\boldsymbol{\sigma}_{k,T} + \boldsymbol{C}\boldsymbol{\xi}, \boldsymbol{\kappa}_{k,T}\right),$$

$$\boldsymbol{T}_{k,T}(\boldsymbol{\xi}) := \boldsymbol{C}\boldsymbol{\xi} - \frac{3\mu}{3\mu + H_m}\sqrt{\frac{2}{3}}\boldsymbol{\Phi}^+\left(\boldsymbol{\sigma}_{k,T} + \boldsymbol{C}\boldsymbol{\xi}, \boldsymbol{\kappa}_{k,T}\right)\overline{\boldsymbol{n}}_{k,T}(\boldsymbol{\xi}),$$

$$\boldsymbol{T}^o_{k,T}(\boldsymbol{\xi}) := \begin{cases} \boldsymbol{C}, \quad \text{if } \boldsymbol{\Phi}\left(\boldsymbol{\sigma}_{k,T} + \boldsymbol{C}\boldsymbol{\xi}, \boldsymbol{\kappa}_{k,T}\right) \le 0, \text{ otherwise} \\ \\ \boldsymbol{C} - 2\mu\frac{3\mu}{3\mu+H_m}\boldsymbol{E}_\varepsilon - \\ -2\mu\frac{3\mu}{3\mu+H_m}\sqrt{\frac{2}{3}}\frac{\sigma_y + H_m\boldsymbol{\kappa}_{k,T}}{\|\boldsymbol{E}_\sigma(\boldsymbol{\sigma}_{k,T}+\boldsymbol{C}\boldsymbol{\xi})\|_\sigma}\left(\overline{\boldsymbol{n}}_{k,T}(\boldsymbol{\xi})\overline{\boldsymbol{n}}^T_{k,T}(\boldsymbol{\xi}) - \boldsymbol{E}_\varepsilon\right), \end{cases}$$

respectively.

Based on the introduced notation we define the non-linear operator $\boldsymbol{F}_k : \mathbb{R}^n \to \mathbb{R}^n$,

$$\boldsymbol{F}_k(\boldsymbol{v}) = \sum_{T\in\mathcal{T}_h}\left(\boldsymbol{T}_{k,T}(\boldsymbol{G}\boldsymbol{R}_T\boldsymbol{v})\right)^T \boldsymbol{G}\boldsymbol{R}_T, \quad \boldsymbol{v} \in \mathbb{R}^n \tag{2.5.48}$$

which represents the left hand side in (2.5.45), and the stiffness matrix $\boldsymbol{K}_k(\boldsymbol{v}) \in \mathbb{R}^{n\times n}$, $\boldsymbol{v} \in \mathbb{R}^n$,

$$\boldsymbol{K}_k(\boldsymbol{v}) = \sum_{T\in\mathcal{T}_h}\left(\boldsymbol{T}^o_{k,T}(\boldsymbol{G}\boldsymbol{R}_T\boldsymbol{v})\boldsymbol{G}\boldsymbol{R}_T\right)^T \boldsymbol{G}\boldsymbol{R}_T, \tag{2.5.49}$$

which represents the bilinear form $a_{k,h}(v_h)$. In particular, we denote the matrix $\boldsymbol{K}_k(\triangle \boldsymbol{u}_{k+1,i})$ briefly by $\boldsymbol{K}_{k,i}$, where the reason of $\triangle \boldsymbol{u}_{k+1,i} \in \mathbb{R}^n$ will be explained in the next section.

Then by using (2.5.48), we can rewrite the equation (2.5.45) as follows: find $\triangle \boldsymbol{u}_{k+1} \in \boldsymbol{V}$ such that

$$\boldsymbol{v}^T \left(\boldsymbol{F}_k(\triangle \boldsymbol{u}_{k+1}) - \triangle \boldsymbol{f}_{k+1} \right) = 0 \quad \forall \boldsymbol{v} \in \boldsymbol{V}, \tag{2.5.50}$$

where $\triangle \boldsymbol{f}_{k+1}$ is the increment of the load vector and $\boldsymbol{V}$ is define in (1.2.24). Let $\tilde{\boldsymbol{u}}_k \in \mathbb{R}^{n-m_U}$, $\tilde{\boldsymbol{f}}_k \in \mathbb{R}^{n-m_U}$, $\tilde{\boldsymbol{K}}_{k,i} \in \mathbb{R}^{(n-m_U)\times(n-m_U)}$, and $\tilde{\boldsymbol{F}}_k : \mathbb{R}^{n-m_U} \to \mathbb{R}^{n-m_U}$ denote the restrictions of $\boldsymbol{u}_k$, $\boldsymbol{f}_k$, $\boldsymbol{K}_{k,i}$, and $\boldsymbol{F}_k$ with respect to the Dirichlet conditions. Then we can rewrite equation (2.5.50) to the following system of non-linear equations:

$$\text{find } \triangle \boldsymbol{u}_{k+1} \in \boldsymbol{V}: \quad \tilde{\boldsymbol{F}}_k(\triangle \tilde{\boldsymbol{u}}_{k+1}) = \triangle \tilde{\boldsymbol{f}}_{k+1}. \tag{2.5.51}$$

The discretized elastoplastic problem can be solved by the following algorithm:

Algorithm 2 (Solution of discretized elasto-plastic problem)**.**

initial step: $\boldsymbol{u}_0 = \boldsymbol{o},\ \boldsymbol{\varepsilon}_{0,T} = \boldsymbol{o},\ \boldsymbol{\sigma}_{0,T} = \boldsymbol{o},\ \boldsymbol{\kappa}_{0,T} = \boldsymbol{o}$ *for any* $T \in \mathcal{T}_h$
for $k = 0,\ \ldots, N-1$ ***do***
  find $\triangle \boldsymbol{u}_{k+1} \in \boldsymbol{V}$*:* $\tilde{\boldsymbol{F}}_k(\triangle \tilde{\boldsymbol{u}}_{k+1}) = \triangle \tilde{\boldsymbol{f}}_{k+1}$
  for all $T \in \mathcal{T}_h$ ***do***
    $\triangle \boldsymbol{\varepsilon}_{k+1,T} = \boldsymbol{G}\boldsymbol{R}_T \triangle \boldsymbol{u}_{k+1},\ \boldsymbol{\varepsilon}_{k+1,T} = \boldsymbol{\varepsilon}_{k,T} + \triangle \boldsymbol{\varepsilon}_{k+1,T}$
    $\triangle \boldsymbol{\sigma}_{k+1,T} = \boldsymbol{T}_{k,T}(\triangle \boldsymbol{\varepsilon}_{k+1,T}),\ \boldsymbol{\sigma}_{k+1,T} = \boldsymbol{\sigma}_{k,T} + \triangle \boldsymbol{\sigma}_{k+1,T}$
    $\triangle \boldsymbol{\kappa}_{k+1,T} = \boldsymbol{T}_{\kappa,k,T}(\triangle \boldsymbol{\varepsilon}_{k+1,T}),\ \boldsymbol{\kappa}_{k+1,T} = \boldsymbol{\kappa}_{k,T} + \triangle \boldsymbol{\kappa}_{k+1,T}$
  end for
end for

2.6 Numerical methods

In this section, we propose efficient methods for the numerical solution of Step 3 of the above algorithm based on the combination of semismooth Newton method and TFETI domain decomposition.

2.6.1 Semismooth Newton method for one time-step problem

The non-linear system of equations (2.5.51) (Step 3 of Algorithm 2) is solved by the semismooth Newton method, see e.g. [53]. The corresponding algorithm is following:

Algorithm 3 (Semismooth Newton method)**.**

initialization: $\triangle \boldsymbol{u}_{k,0} = \boldsymbol{o}$

2: ***for*** $i = 0, 1, 2, \ldots$ ***do***
3: *find* $\boldsymbol{\delta u}_i \in \boldsymbol{V}$: $\tilde{\boldsymbol{K}}_{k,i}\boldsymbol{\delta}\tilde{\boldsymbol{u}}_i = \triangle\tilde{\boldsymbol{f}}_{k+1} - \tilde{\boldsymbol{F}}_k(\triangle\tilde{\boldsymbol{u}}_{k,i})$
4: *compute* $\triangle\boldsymbol{u}_{k,i+1} = \triangle\boldsymbol{u}_{k,i} + \boldsymbol{\delta u}_i$
5: **if** $\|\triangle\boldsymbol{u}_{k,i+1} - \triangle\boldsymbol{u}_{k,i}\|/(\|\triangle\boldsymbol{u}_{k,i+1}\| + \|\triangle\boldsymbol{u}_{k,i}\|) \leq \epsilon_{Newton}$ **then stop**
6: ***end for***
7: *set* $\triangle\boldsymbol{u}_{k+1} = \triangle\boldsymbol{u}_{k,i+1}$

Here $\epsilon_{Newton} > 0$ is the relative stopping tolerance and $\boldsymbol{\delta}\tilde{\boldsymbol{u}}_{\boldsymbol{i}} \in \mathbb{R}^{n-m_U}$ is the restriction of $\boldsymbol{\delta u_i}$ given by omitting the entries (degrees of freedom) corresponding to the prescribed Dirichlet conditions. The systems of linear equations, which are considered in each Newton iteration, are solved in parallel by the Total-FETI method introduced in the next subsection.

In [10, 35, 61], superlinear local convergence of the algorithm has been derived. Let us note that the convergence depends on the discretization parameter h of the triangulation. Therefore we can expect that the finer mesh the bigger number of the Newton iterations. In [61], a damped semismoooth Newton method for similar problems has also been described. Such a method has again superlinear local convergence and additionally global convergence.

For the Drucker-Prager model with perfect plasticity we use instead of matrix $\boldsymbol{K}_{k,i}$ the matrix $\boldsymbol{K}$, representing the elastic stiffness matrix to ensure more reliable convergence.

2.6.2 TFETI based algorithms for solving elasto-plastic problem

In this subsection, we summarize the algorithm for solving the whole elasto-plastic problem based on the semismooth Newton method introduced in Subsection 2.6.1 and the TFETI domain decomposition method introduced in Section 1.3. This is possible, since the matrices $\tilde{\mathbf{K}}_{k,i}$ are again symmetric and positive definite for the von Mises criterion with the investigated hardening models.

Algorithm 4 (TFETI based algorithm for solving elastoplastic problem - sequential version)**.**

1: $\mathbf{u}_0 = \mathbf{o}, \boldsymbol{\sigma}_{0,T} = \mathbf{o}, \boldsymbol{\kappa}_{0,T} = \mathbf{o},\ T \in \mathcal{T}_h$ *(initial step)*
2: ***for*** $k = 0,\ 1,\ 2,\ \ldots,\ N-1$ *(time steps)* ***do***
3: *set* $\triangle\mathbf{u}_{k+1,0} = \mathbf{o}$ *(zero approximation)*
4: ***for*** $i = 1, 2, \ldots$ *(Newton iterations)* ***do***
5: ***for*** $p = 1, 2, \ldots, s$ *(cycle over subdomains)* ***do***
6: *restrict* $\triangle\mathbf{u}_{k+1,i-1}, \triangle\mathbf{f}_{k+1}$ *into subdomain variables* $\triangle\boldsymbol{u}^p_{k+1,i-1}$, $\triangle\boldsymbol{f}^p_{k+1}$
7: *call Algorithm 5 with* $(\triangle\boldsymbol{u}^p_{k+1,i-1}, \triangle\boldsymbol{f}^p_{k+1}, \boldsymbol{\sigma}_{k,T}, \boldsymbol{\kappa}_{k,T}, T \in \mathcal{T}^p_h)$ *to find output variables* $\boldsymbol{K}^p_{k,i}$, $\boldsymbol{f}^p_{k,i}$, $\triangle\boldsymbol{\sigma}_{k+1,T}$, $\triangle\boldsymbol{\kappa}_{k+1,T}$, $T \in \mathcal{T}^p_h$.
8: *collect* $\boldsymbol{K}^p_{k,i}\, \boldsymbol{f}^p_{k,i}$, *into global variables* $\mathbf{K}_{k,i}$, $\mathbf{f}_{k,i}$

9: ***end for*** *(cycle over subdomains)*

10: *solve by Algorithm 1 the problem: find* $\boldsymbol{\delta}\mathbf{u}_i \in \mathbf{V}$ *such that*

$$\mathbf{J}_{k,i}(\boldsymbol{\delta}\mathbf{u}_i) \leq \mathbf{J}_{k,i}(\mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{V}, \text{ with } \mathbf{J}_{k,i}(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{K}_{k,i}\mathbf{v} - \mathbf{f}_{k,i}^T\mathbf{v}$$

11: $\triangle\mathbf{u}_{k+1,i} = \triangle\mathbf{u}_{k+1,i-1} + \boldsymbol{\delta}\mathbf{u}_i$ *(displacement update)*

12: **if** $\|\triangle\mathbf{u}_{k+1,i} - \triangle\mathbf{u}_{k+1,i-1}\| / \left(\|\triangle\mathbf{u}_{k+1,i}\| + \|\triangle\mathbf{u}_{k+1,i-1}\|\right) \leq \epsilon_{Newton}$ **then stop**

13: ***end for****(Newton iter.)*

14: $\mathbf{u}_{k+1} = \mathbf{u}_k + \triangle\mathbf{u}_{k+1,i}$,

15: $\boldsymbol{\sigma}_{k+1,T} = \boldsymbol{\sigma}_{k,T} + \triangle\boldsymbol{\sigma}_{k+1,T}$, $\boldsymbol{\kappa}_{k+1,T} = \boldsymbol{\kappa}_{k,T} + \triangle\boldsymbol{\kappa}_{k+1,T}$, $T \in \mathcal{T}_h$

16: ***end for*** *(cycle over time steps)*

The assembling procedure for subdomain data looks as follows.

Algorithm 5 (Assemble all data corresponding to a subdomain Ω^p)**.**

1: *Input:* $\triangle\boldsymbol{u}^p_{k+1,i-1}$, $\triangle\boldsymbol{f}^p_{k+1}$, $\boldsymbol{\sigma}_{k,T}$, $\boldsymbol{\kappa}_{k,T}$, $T \in \mathcal{T}_h^p$.

2: $\boldsymbol{f}^p_{k,i} = \triangle\boldsymbol{f}^p_{k+1}$

3: $\boldsymbol{K}^p_{k,i} = \boldsymbol{O}$

4: ***for*** $T \in \mathcal{T}_h^p$ ***do***

5: *compute* $|T|$ *(volume of the element* T*)*

6: $\triangle\boldsymbol{\sigma}_{k+1,T} = \boldsymbol{T}_{k,T}\left(\boldsymbol{GR}^p_T \triangle\boldsymbol{u}^p_{k+1,i-1}\right)$

7: $\triangle\boldsymbol{\kappa}_{k+1,T} = \boldsymbol{T}_{\kappa,k,T}\left(\boldsymbol{GR}^p_T \triangle\boldsymbol{u}^p_{k+1,i-1}\right)$

8: $\boldsymbol{f}^p_{k,i} = \boldsymbol{f}^p_{k,i} - |T|\left(\triangle\boldsymbol{\sigma}_{k,T}\right)^T \boldsymbol{GR}^p_T$

9: $\boldsymbol{K}^p_{k,i} = \boldsymbol{K}^p_{k,i} + |T|\left(\boldsymbol{T}^o_{k,T}\left(\boldsymbol{GR}^p_T \triangle\boldsymbol{u}^p_{k+1,i-1}\right)\boldsymbol{GR}^p_T\right)^T \boldsymbol{GR}^p_T$

10: ***end for*** *(cycle over elements)*

11: *Output:* $\boldsymbol{K}^p_{k,i}$, $\boldsymbol{f}^p_{k,i}$, $\triangle\boldsymbol{\sigma}_{k+1,T}, \triangle\boldsymbol{\kappa}_{k+1,T}$, $T \in \mathcal{T}_h^p$

Loop over subdomains and all subdomain operations may be implemented in parallel. Parallelization of FETI/TFETI is based on distributing matrix portions among processing units. This allows algorithms to be almost the same in sequential and parallel versions; only data structure implementation differs. Most of computations (subdomain operations) appearing are purely local and therefore parallelizable without any data transfers. Each of cores works with the local part associated with its subdomains. Natural effort using the massively parallel computers is to maximize the number of subdomains so that the sizes of the subdomain stiffness matrices are reduced. This accelerates their factorization and subsequent pseudoinverse application which belongs to the most time consuming action.

On the other hand, negative effect of that is an increase of the null space dimension and the number of Lagrange multipliers. This leads to larger coarse problems, i.e., applications of the projector $\mathbf{P_N}$ which are of negligible cost only up to a thousand of cores and then the coarse problem solution starts to dominate. For the numerical solution of such large problems we recommend to use a hybrid FETI method.

2.7 Comments to plane strain problem for elasto-plasticity

In Section 1.4 we introduced the assumptions and differences for the case of plane strain. Similarly, a dimensional reduction based on plane strain can be considered for elasto-plasticity. We assume that the investigated plastic criteria remain the same as for a general 3D model, see e.g. [35]. It means that we must actively use the stress component σ_{33} in comparison to elasticity. Moreover it follows from the plastic flow rules that the component ε^p_{33} of the plastic strain need not be generally equal to zero.

Implementation details can be found in [35].

2.8 Numerical experiments

Let us consider the same benchmarks as in Chapter 1 and show differences between elastic and elasto-plastic material models. Since the elasto-plastic models are rate-independent and any local unloading is not expected for the benchmarks with respect to the prescribed load history, the results should be independent of the chosen time discretization. The benchmark introduced in Subsection 1.5.1 we resolve for von Mises model with both isotropic and kinematic hardening. To compare them we consider the equivalent setting of elasto-plastic parameters. The benchmark in Subsection 1.5.2 is resolved for Drucker-Prager criterion with perfect plasticity. The stopping tolerances of the Newton and the PCGP algorithms are

$$\epsilon_{Newton} = 10^{-5} \text{ and } \epsilon_{PCGP} = 10^{-9}, \tag{2.8.52}$$

respectively.

2.8.1 The beam with a hole with von Mises criterion and isotropic hardening

Let us consider the same benchmark as in Subsection 1.5.1. The plastic material parameters are $\sigma_y = 450,\ H_m = 100$. The indicated traction forces in Figure 1.4 with the history of loading taking into account are prescribed by the function

$$g(t) = 450\sin(2\pi t),\ t \in [t_0, t^*],\ t_0 = 0,\ t^* = \frac{1}{4}. \tag{2.8.53}$$

Let us consider two variants of the equidistant time discretization characterized by the time step Δt:

(*a*) $\Delta t = 1/4$, $N = 1$,

(*b*) $\Delta t = 1/48$, $N = 12$.

In Table 2.1, the number of the PCGP iterations, the number of plastic elements, and the relative convergence criterion are reported for each Newton's iteration of Algorithm 4 and the time discretization (*a*). In this case, we observe quadratic convergence of the Newton method after identification of the plastic zone in iteration 6. Such behavior agrees with the theoretical results in [10].

The computational history of Algorithm 4 for the time discretization (*b*) and the stopping tolerances (2.8.52) is documented in Table 2.2. The corresponding development of the plastic zone is depicted for the times t_3, t_6, t_9, and t_{12} in Figures 2.1 - 2.4, respectively. The growing zone of plastic elements results from the monotonically increased loading. Distributions of the von Mises stress $\|\mathrm{dev}(\sigma)\|_F$ and the total displacement $\|u\|$ at the final time t^* are depicted in Figures 2.5 and 2.6, respectively.

Newton iter.	Number of PCGP iters.	Number of plastic elem.	Stopping criterion
1	100	0	1
2	125	7 308	9.9750e-2
3	126	6 788	6.2381e-2
4	136	6 791	1.7253e-2
5	134	6 933	1.5295e-3
6	134	6 963	4.5745e-5
7	133	6 963	3.7495e-8

Table 2.1: Computational history of Algorithm 4 for the time discretization (*a*) and von Mises criterion with isotropic hardening

Comparing time discretizations (*a*) and (*b*) we see that the resulting number of plastic elements differs at the final time t^*. The difference is less than 0.5% and is caused by the numerical errors, the use of iterative solvers, and the numerical evaluation of the yield function which decides whether an element plasticizes or not.

Comparing Figures 2.5 and 1.6 we see that the von Mises stress in Figure 1.6 is much larger than in Figure 2.5. This is expected phenomenon. Opposite situation is in Figures 2.6 and 1.7, where total displacements are depicted. Such behaviour follows from properties of hysteresis curves for elastic and elasto-plastic problems.

2.8.2 The beam with a hole with von Mises criterion and kinematic hardening

Let us consider the same example as above but with kinematic hardening instead of isotropic one. To compare the results of models with isotropic and kinematic hardening we set $\sigma_y = 450$ and $c_0 = \frac{2}{3}H_m = 66.67$. The indicated traction forces in Figure 1.4 are

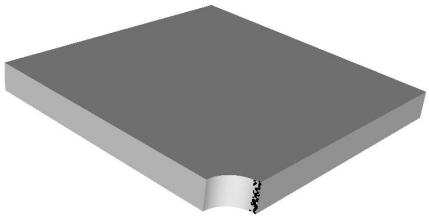

Figure 2.1: Elastic (gray color) and plastic (black color) elements at time t_3

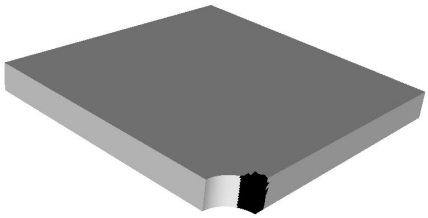

Figure 2.2: Elastic (gray color) and plastic (black color) elements at time t_6

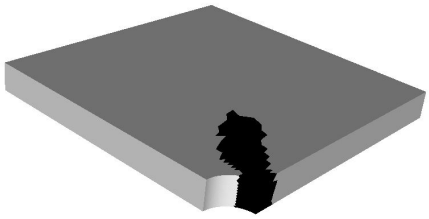

Figure 2.3: Elastic (gray color) and plastic (black color) elements at time t_9

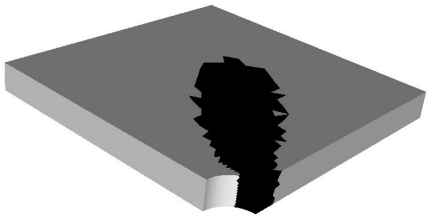

Figure 2.4: Elastic (gray color) and plastic (black color) elements at time t_{12}

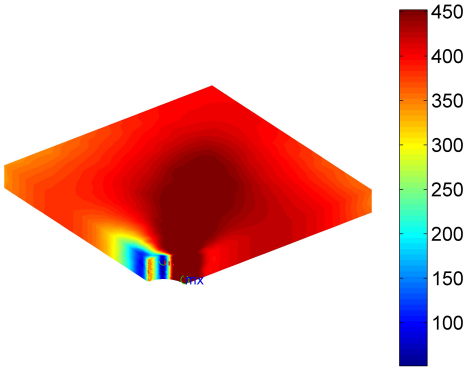

Figure 2.5: Distribution of von Mises stress at t_{12}

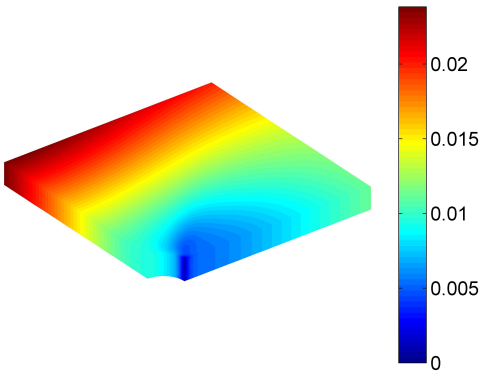

Figure 2.6: Total displacement at t_{12}

Time step	Number of Newton iters.	Total number of PCGP iterations	Number of plastic elems.
1	2	213	0
2	2	216	0
3	3	322	113
4	5	507	1 108
5	5	497	1 848
6	4	399	2 452
7	4	420	3 284
8	5	588	4 573
9	5	636	5 339
10	5	688	6 103
11	5	765	6 755
12	4	684	6 981

Table 2.2: Computational history of Algorithm 4 for the time discretization (b) and von Mises criterion with isotropic hardening

prescribed by (2.8.53). Let us consider two variants of the equidistant time discretization characterized by the time step Δt:

(a) $\Delta t = 1/4$, $N = 1$,

(b) $\Delta t = 1/64$, $N = 16$.

We report the number of the PCGP iterations, the number of plastic elements, and the relative convergence criterion for each Newton's iteration of Algorithm 4 and the time discretization (a) in Table 2.3. When we compare it with Table 2.1 we see that they are almost the same because of the choice of c_0.

The computational history of Algorithm 4 for the time discretization (b) and the stopping tolerances (2.8.52) is documented in Table 2.4. The corresponding development of the plastic zone is depicted for the times t_4, t_8, t_{12}, and t_{16} in Figures 2.7 - 2.10, respectively. Distributions of the von Mises stress $\|\mathrm{dev}(\sigma)\|_F$ and the total displacement $\|u\|$ at the final time t^* are depicted in Figures 2.11 and 2.12, respectively. Comparing the achieved results with those computed in the previous example we see immediately that both hardening models give "the same" results for our benchmarks up to numerical errors.

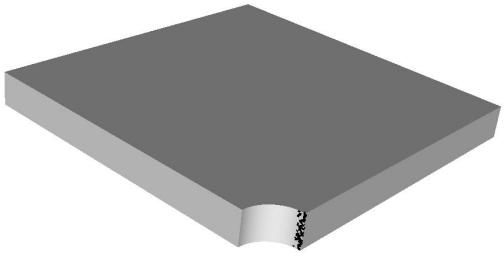

Figure 2.7: Elastic (gray color) and plastic (black color) elements at time t_4

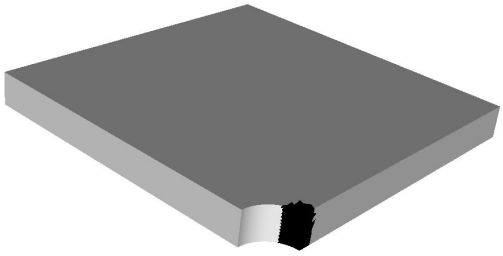

Figure 2.8: Elastic (gray color) and plastic (black color) elements at time t_8

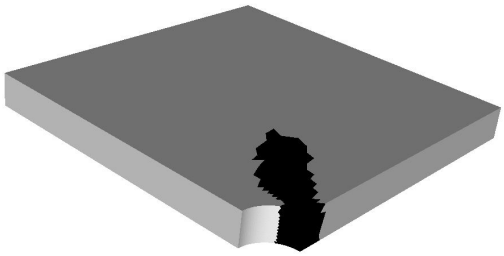

Figure 2.9: Elastic (gray color) and plastic (black color) elements at time t_{12}

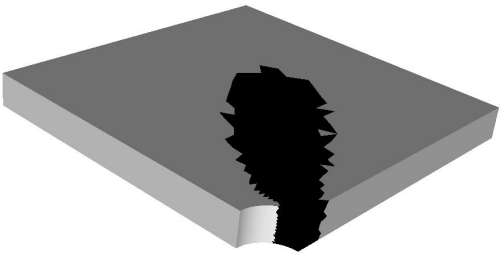

Figure 2.10: Elastic (gray color) and plastic (black color) elements at time t_{16}

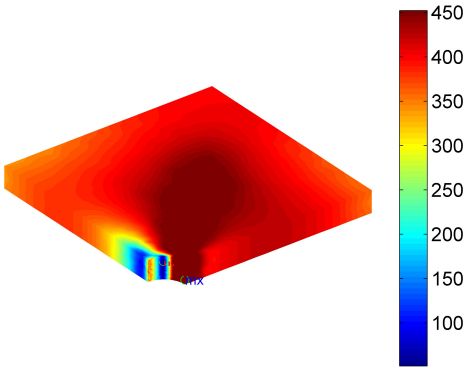

Figure 2.11: Distribution of von Mises stress at t_{16}

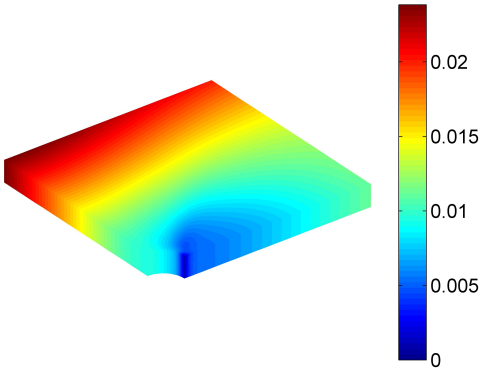

Figure 2.12: Total displacement at t_{16}

Newton iter.	Number of PCGP iters.	Number of plastic elem.	Stopping criterion
1	100	0	1
2	123	7 308	9.9750e-2
3	127	6 788	6.2381e-2
4	137	6 791	1.7253e-2
5	133	6 933	1.5295e-3
6	132	6 963	4.5750e-5
7	134	6 963	5.9600e-8

Table 2.3: Computational history of Algorithm 4 for the time discretization (a) and von Mises criterion with kinematic hardening

2.8.3 Äspö Pillar Stability Experiment for Drucker-Prager criterion with perfect plasticity

Let us consider the same benchmark as in Subsection 1.5.2 and apply Drucker-Prager material model. The material parameters are

$$\overline{c} = c\frac{3}{\sqrt{9 + 12\,\tan^2\phi}} \quad \text{and} \quad \eta = \frac{3\,\tan\phi}{\sqrt{9 + 12\,\tan^2\phi}},$$

where $c = 30MPa$, $\phi = 49Deg$. The indicated traction forces in Figure 1.8 with the history of loading taking into account are prescribed by the functions

$$g_1(t) = 13 \cdot 10^6 t,\ g_2(t) = 43 \cdot 10^6 t,\ t \in [t_0, t^*],\ t_0 = 0,\ t^* = 1.$$

Let us consider two variants of the equidistant time discretization characterized by the time step Δt:

(a) $\Delta t = 1$, $N = 1$,

(b) $\Delta t = 1/4$, $N = 4$.

Let us recall that we use the elastic stiffness matrix in each Newton iteration to stabilize the convergence. Therefore it is not quadratic but only linear.

In Table 2.5, the number of the PCGP iterations, the number of plastic elements, and the relative convergence criterion are reported for each Newton's iteration of Algorithm 4 and the time discretization (a).

On the other hand computational history of Algorithm 4 for the time discretization (b) and the stopping tolerances (2.8.52) is documented in Table 2.6. From Tables 2.5 and 2.6 we see that the number of plastic elements are the same.

In Figure 2.13, we depict the total displacement. Distributions of the von Mises stress $\|\mathrm{dev}(\sigma)\|_F$, normal stresses σ_{xx}, σ_{yy}, and shear stress σ_{xy} after loading are depicted in Figures 2.14, 2.15, 2.16, and 2.17, respectively.

The differences between these figures and those in Subsection 1.5.2 are very small. The biggest difference is visible comparing Figures 2.17 and 1.14.

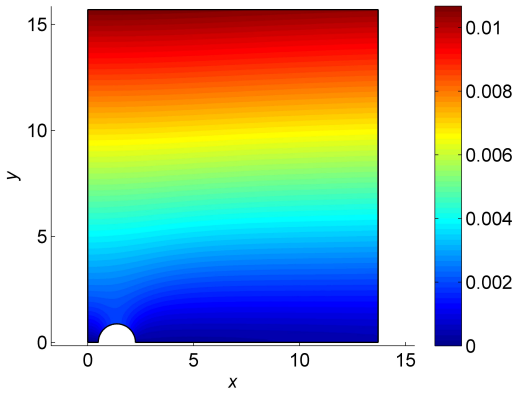

Figure 2.13: Total displacement

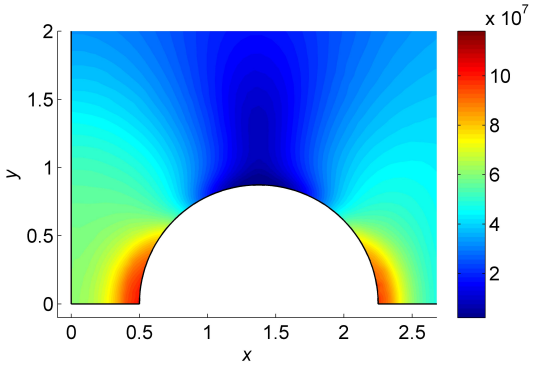

Figure 2.14: Distribution of von Mises stress

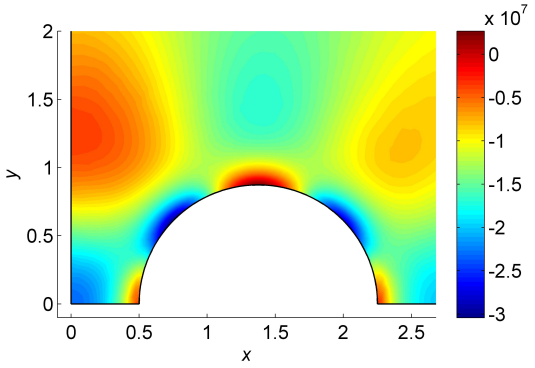

Figure 2.15: Distribution of stress σ_{xx}

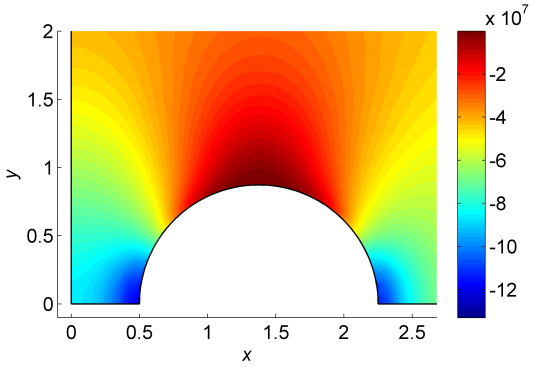

Figure 2.16: Distribution of stress σ_{yy}

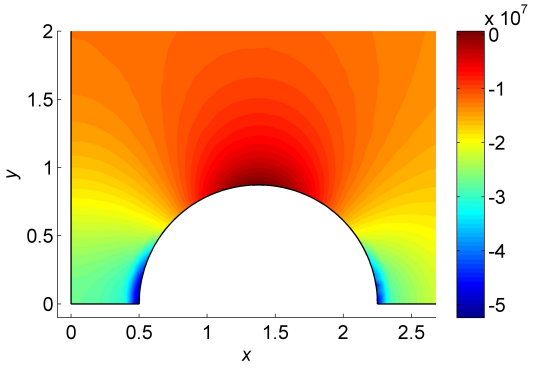

Figure 2.17: Distribution of stress σ_{xy}

Time step	Number of Newton iters.	Total number of PCGP iterations	Number of plastic elems.
1	2	218	0
2	2	218	0
3	2	219	0
4	3	327	114
5	4	420	866
6	4	405	1 513
7	4	403	2 005
8	4	406	2 456
9	4	414	3 041
10	5	563	4 067
11	4	491	4 809
12	5	650	5 341
13	5	695	5 915
14	5	754	6 476
15	5	813	6 850
16	4	712	6 987

Table 2.4: Computational history of Algorithm 4 for the time discretization (b) and von Mises criterion with kinematic hardening

2.9 Scalability experiments

In this section we show the parallel and "the numerical scalability" tests for 2D and 3D benchmarks. We resolve the problem with varying discretizations and decompositions defined by the discretization parameter h and the decomposition parameter H, respectively. For each h and H, the bodies are discretized by regular mesh with triangular or hexahedral elements and decomposed into the subdomains. For sake of brevity we assume only two cases: 1. elasto-plastic square with von Mises criterion and isotropic hardening and 2. elasto-plastic beam with von Mises criterion and kinematic hardening.

The proposed algorithms were parallelized using Matlab Distributed Computing Server and Matlab Parallel Toolbox. For all computations we use 33 cores with 2GB memory per core of the HP Blade system, model BLc7000. The stopping tolerances of the Newton and the PCGP algorithms are

$$\epsilon_{Newton} = 10^{-4} \text{ and } \epsilon_{PCGP} = 10^{-7},$$

respectively.

Newton iter.	Number of PCGP iters.	Number of plastic elem.	Stopping criterion
1	122	0	1
2	123	105	1.3619e-4
3	123	108	1.0024e-4
4	122	109	7.6447e-5
5	122	109	5.9701e-5
6	122	109	4.7436e-5
7	123	110	3.8198e-5
8	122	110	3.1081e-5
9	122	110	2.5503e-5
10	123	111	2.1073e-5
11	123	111	1.7530e-5
12	122	111	1.4649e-5
13	123	111	1.2294e-5
14	122	111	1.0356e-5
15	123	111	8.7532e-6

Table 2.5: Computational history of Algorithm 4 for the time discretization (a) and Drucker-Prager criterion with perfect plasticity

Time step	Number of Newton iters.	Total number of PCGP iterations	Number of plastic elems.
1	2	255	0
2	2	255	0
3	9	1 151	29
4	22	2 814	111

Table 2.6: Computational history of Algorithm 4 for the time discretization (b) and Drucker-Prager criterion with perfect plasticity

2.9.1 Parallel scalability for von Mises criterion with isotropic hardening in 2D

In this benchmark we consider an elasto-plastic square of size 10×10. Its geometry with imposed boundary conditions is depicted in Figure 2.18. A similar benchmark was solved in [64].

The elasto-plastic body Ω is made of homogenous isotropic material with the parameters: $E = 206\,900$, $\nu = 0.29$, $H_m = 100$, $\sigma_y = 450$. The indicated traction forces with the history of loading taking into account are prescribed by

$$g(t) = 450 \sin(2\pi t),\ t \in [t_0, t^*],\ t_0 = 0,\ t^* = \frac{1}{4}.$$

Let us focus only on one time step problem. For the spatial discretization of Ω, let

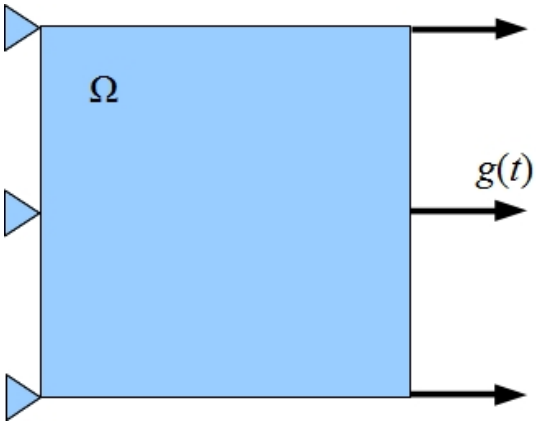

Figure 2.18: Geometry of Ω

us consider regular triangular mesh generated by MatSol with 103 041 nodes and 204 800 elements decomposed into 256 subdomains. The number of primal variables is 225 792 and the number of dual variables is 20 352.

In Table 2.7 and Figure 2.19 we document the parallel scalability of the proposed algorithm. We see that the number of the Newton iterations, number of plastic elements, and total number of PCGP iterations remain constant for all cases. In Figures 2.20 and 2.21, we depict the von Mises stress distribution and the total displacement for 103 041 nodes and 204 800 elements.

Number of cores	2	3	5	9	17	33
Number of plastic elems.	978	978	978	978	978	978
Number of Newton iters.	5	5	5	5	5	5
Total number of PCGP iters.	270	270	270	270	270	270
Total time [sec]	431.6	227.6	129.5	92.6	69.1	49.5

Table 2.7: Paralel scalability of Algorithm 4 for different number of cores in 2D.

In Table 2.8 we report "the numerical scalability" for different mesh levels. The most important is row with total number of PCGP iterations, where we can see, that the number of iterations grows only moderately.

2.9.2 Parallel scalability for von Mises criterion with kinematic hardening in 3D

In this benchmark we consider an elasto-plastic homogenous beam of size $8 \times 4 \times 4$. Its geometry with imposed boundary conditions is depicted in Figure 2.22.

The elasto-plastic body Ω is made of homogenous isotropic material with the same parameters as in the previous benchmark ($c_0 = 66.7$). The indicated traction forces in

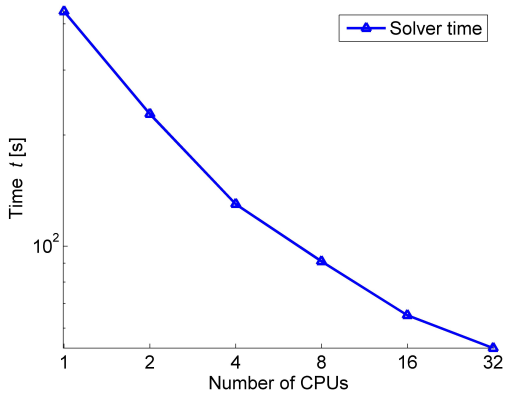

Figure 2.19: Parallel scalability (2D case)

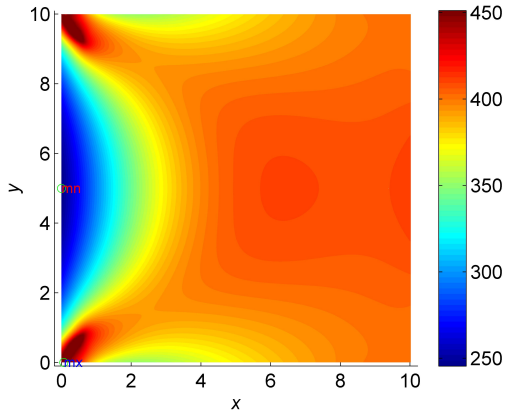

Figure 2.20: Distribution of von Mises stress

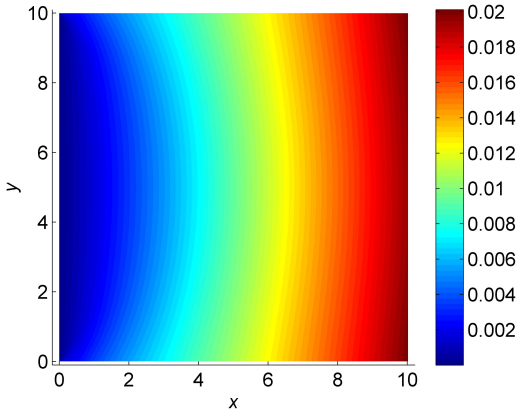

Figure 2.21: Total displacement

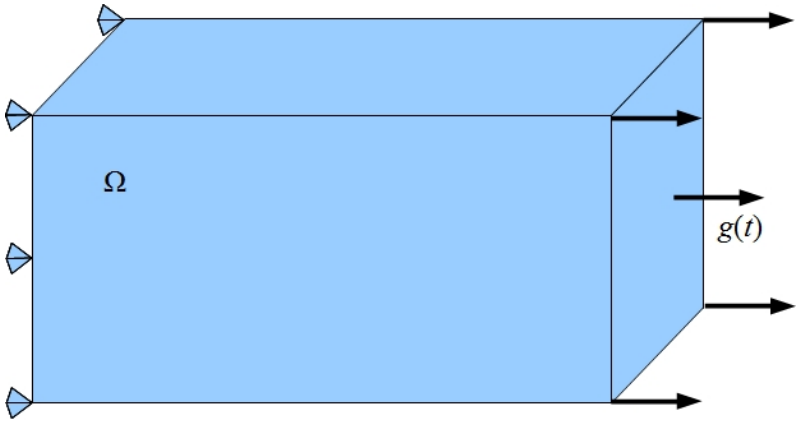

Figure 2.22: Geometry of Ω

Mesh level	1	2	3	4	5	6
Mesh nodes	441	1 601	6 561	25 921	103 041	410 881
Mesh elements	800	3 200	12 800	51 200	204 800	819 200
Number of subdomains	1	4	16	64	256	1 024
Number of cores	2	3	9	17	33	33
Primal variables	1 014	3 528	14 112	56 448	225 792	903 168
Dual variables	174	248	1 152	4 928	20 352	82 688
Number of plastic elems.	4	18	82	280	978	3 734
Number of Newton iters.	3	4	5	5	5	5
Total number of PCGP iters.	44	128	202	219	270	411
Total time [sec]	1.2	2.9	4.7	11.3	49.3	360.3

Table 2.8: Performance of Algorithm 4 for different mesh levels in 2D.

Figure 2.22 with the history of loading taking into account are prescribed by

$$g(t) = 400 \sin(2\pi t),\ t \in [t_0, t^*],\ t_0 = 0,\ t^* = \frac{1}{4}.$$

Let us focus again only on one time step problem. For the spatial discretization of Ω, let us consider hexahedron mesh generated by MatSol with 136 161 nodes and 128 000 elements decomposed into 128 subdomains. The number of primal variables is 511 104 and the number of dual variables is 107 664.

In Table 2.9 and Figure 2.23 we document the parallel scalability of the proposed algorithm. We see that the number of the Newton iterations, number of plastic elements, and total number of PCGP iterations remain constant for all cases. In Figures 2.24 and 2.25, we depict the von Mises stress distribution and the total displacement.

Number of cores	2	3	5	9	17	33
Number of plastic elems.	2104	2104	2104	2104	2104	2104
Number of Newton iters.	3	3	3	3	3	3
Total number of PCGP iters.	96	96	96	96	96	96
Total time [sec]	1099.8	652.4	354.9	215.1	141.8	106.4

Table 2.9: Paralel scalability of Algorithm 4 for different number of cores in 3D.

In Table 2.10 we report "the numerical scalability" for different mesh levels. The most important is the row with total number of PCGP iterations, where we can see again, that the number of iterations grows only moderately.

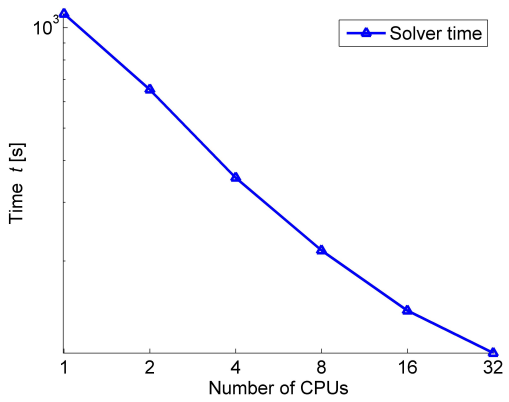

Figure 2.23: Parallel scalability (3D case)

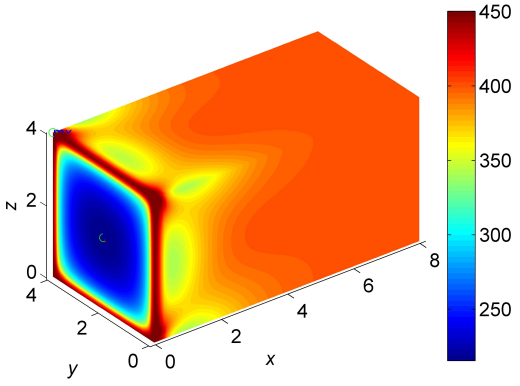

Figure 2.24: Distribution of von Mises stress

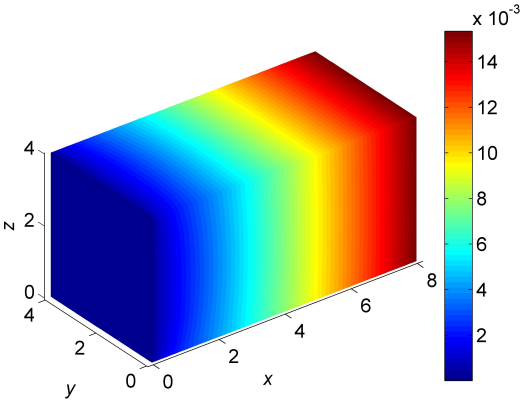

Figure 2.25: Total displacement

Mesh level	1	2	3	4
Mesh nodes	2 541	18 081	136 161	1 056 321
Mesh elements	2 000	16 000	128 000	1 024 200
Number of subdomains	2	16	128	1 024
Number of cores	3	17	33	33
Primal variables	7 986	63 888	511 104	4 088 832
Dual variables	726	10 968	107 664	939 552
Number of plastic elems.	16	352	2 104	16 944
Number of Newton iters.	2	3	3	3
Total number of PCGP iters.	33	90	96	115
Total time [sec]	6.0	14.5	105.9	2350.9

Table 2.10: Performance of Algorithm 4 for different mesh levels in 3D.

Chapter 3

Elasticity with contact

In this chapter we deal with elastic multibody contact problems without friction. More precisely, we consider contact of o - homogenous isotropic elastic bodies, where $o \geq 2$. Each body occupies a domain Ω^d with a sufficiently smooth boundary Γ^d, $d = 1, 2, \ldots, o$. The boundary Γ^d, in contrast to elastic problem without contact, is divided into 3 mutually disjoint parts Γ_U^d, Γ_N^d, and Γ_C^d. The part Γ_C^d describes a possible contact boundary of domain Ω^d with other bodies, it means $\Gamma_C^d = \bigcup_{l \neq d} \Gamma_C^{dl}$, where Γ_C^{dl} is a part of boundary Γ_C^d, which can be in contact with body Ω^l. Furthermore, we assume the linearized non-interpenetration conditions, which can be found for example in [43, 67].

For the sake of brevity, in this chapter, we consider only algebraic formulation of the contact problem. Since the problem is solved by the TFETI domain decomposition method, we start with the generalized minimization formulation (1.3.34). We introduce the primal minimization problem and we derive the corresponding dual formulation. Then we denote briefly preconditioners and solvers SMALSE-M and MPGP introduced by Dostal and Kozubek [22] and Dostal and Kucera [25]. In the end of this chapter we show numerical experiments, which are discussed more in Chapter 4.

3.1 TFETI domain decomposition method

In this section we follow the TFETI steps described in Section 1.3. We apply the TFETI domain decomposition method to each body Ω^d and get s_d subdomains $\Omega^{d,1}, \ldots, \Omega^{d,s_d}$, see Figure 3.1. Again we obtain in the resulting minimization problem the equality constraints describing the Dirichlet boundary condition and the gluing conditions. In addition, we get the inequality constraints representing the non-interpenetration conditions in contact zones. The scheme of the corresponding algebraic formulation is following:

$$\min_{\mathbf{v} \in \mathbb{R}^n} \mathbf{J}(\mathbf{v}) \quad \text{subject to} \quad \mathbf{B}_I \mathbf{v} \leq \mathbf{c}_I \quad \text{and} \quad \mathbf{B}_E \mathbf{v} = \mathbf{o}, \tag{3.1.1}$$

with

$$\mathbf{J}(\mathbf{v}) = \frac{1}{2} \mathbf{v}^T \mathbf{K} \mathbf{v} - \mathbf{f}^T \mathbf{v},$$

where $\mathbf{v} \in \mathbb{R}^{\mathtt{n}}$ has the structure

$$\mathbf{v} = \left(\left(\boldsymbol{v}^{1,1}\right)^T, \ldots, \left(\boldsymbol{v}^{1,s_1}\right)^T, \ldots, \left(\boldsymbol{v}^{d,1}\right)^T, \ldots, \left(\boldsymbol{v}^{d,s_d}\right)^T, \ldots, \left(\boldsymbol{v}^{o,1}\right)^T, \ldots, \left(\boldsymbol{v}^{o,s_o}\right)^T \right)^T .$$

Similar structures have also the matrix $\mathbf{K} \in \mathbb{R}^{\mathtt{n}\times\mathtt{n}}$ and the vector $\mathbf{f} \in \mathbb{R}^{\mathtt{n}}$. The minimization formulation (3.1.1) differs from the formulation (1.3.34) due to non-interpenetration condition representing by the matrix $\mathbf{B}_I \in \mathbb{R}^{\mathtt{m}_C\times\mathtt{n}}$ and the vector $\mathbf{c}_I \in \mathbb{R}^{\mathtt{m}_C}$. The matrix $\mathbf{B}_E \in \mathbb{R}^{\mathtt{m}_E\times\mathtt{n}}$ represents the Dirichlet boundary conditions and the gluing conditions. A solution of the problem (3.1.1) is denoted by $\mathbf{u}$. Existence and uniqueness of the solution are discussed for example in [38].

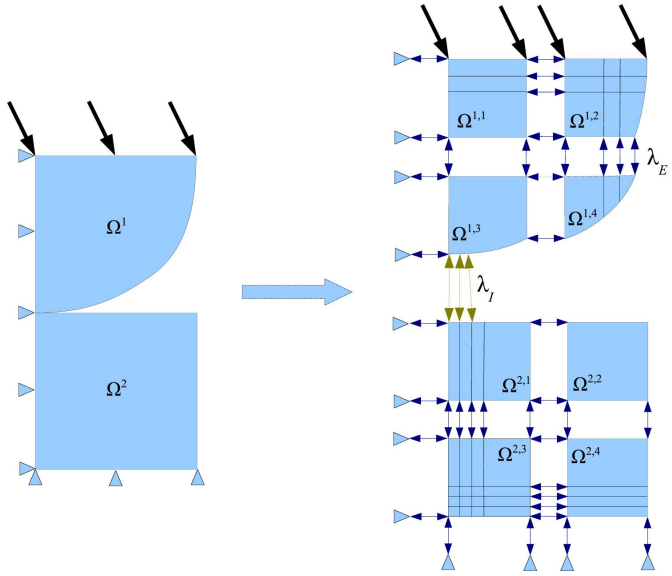

Figure 3.1: TFETI domain decomposition

The constraints in the minimization formulation (3.1.1) can be represented by the convex set

$$\mathcal{K} := \{\mathbf{v} \in \mathbb{R}^{\mathtt{n}} \mid \mathbf{B}_I\mathbf{v} \leq \mathbf{c}_I \ , \ \mathbf{B}_E\mathbf{v} = \mathbf{o}\} . \tag{3.1.2}$$

The problem (3.1.1) is not suitable for numerical solution, since the stiffness matrix $\mathbf{K}$ is typically ill-conditioned and singular, and the feasible set $\mathcal{K}$ is in general so complex that projections into it can hardly be effectively computed. Under these circumstances, it would be very difficult to achieve fast identification of the active set at the solution and fast solution of the auxiliary problems.

The complications mentioned above may be essentially reduced by applying the duality theory of convex programming (see, e.g., Bazaraa, Shetty, and Sherali [8]). In the dual formulation of problem (3.1.1), we use two types of Lagrange multipliers, namely $\boldsymbol{\nu}_I \in \mathbb{R}^{\mathtt{m}_C}$, $\boldsymbol{\nu}_I \geq \mathbf{o}$ related to the non-interpenetration condition, $\boldsymbol{\nu}_E \in \mathbb{R}^{\mathtt{m}_E}$ related to the "gluing" and Dirichlet conditions. The Lagrangian associated with problem (3.1.1) reads

as

$$L(\mathbf{v}, \boldsymbol{\nu}_I, \boldsymbol{\nu}_E) = \frac{1}{2}\mathbf{v}^T\mathbf{K}\mathbf{v} - \mathbf{f}^T\mathbf{v} + \boldsymbol{\nu}_I^T(\mathbf{B}_I\mathbf{v} - \mathbf{c}_I) + \boldsymbol{\nu}_E^T\mathbf{B}_E\mathbf{v}. \tag{3.1.3}$$

Using the convexity of the cost function and constraints, we can use the classical duality theory [8] to reformulate problem (3.1.1) to get

$$\min_{\mathbf{v}\in\mathbb{R}^{\mathtt{n}}} \sup_{\substack{\boldsymbol{\nu}_E\in\mathbb{R}^{\mathtt{m}_E} \\ \boldsymbol{\nu}_I\geq\mathbf{o}}} L(\mathbf{v}, \boldsymbol{\nu}_I, \boldsymbol{\nu}_E) = \max_{\substack{\boldsymbol{\nu}_E\in\mathbb{R}^{\mathtt{m}_E} \\ \boldsymbol{\nu}_I\geq\mathbf{o}}} \min_{\mathbf{v}\in\mathbb{R}^{\mathtt{n}}} L(\mathbf{v}, \boldsymbol{\nu}_I, \boldsymbol{\nu}_E). \tag{3.1.4}$$

To simplify the notation, we denote

$$\boldsymbol{\nu} = \begin{bmatrix} \boldsymbol{\nu}_E \\ \boldsymbol{\nu}_I \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{B}_E \\ \mathbf{B}_I \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} \mathbf{o} \\ \mathbf{c}_I \end{bmatrix},$$

and

$$\Lambda = \{(\boldsymbol{\nu}_E^T, \boldsymbol{\nu}_I^T)^T \in \mathbb{R}^{\mathtt{m}_E+\mathtt{m}_C} : \ \boldsymbol{\nu}_I \geq \mathbf{o}\}.$$

Now we can write the Lagrangian (3.1.3) briefly as

$$L(\mathbf{v}, \boldsymbol{\nu}) = \frac{1}{2}\mathbf{v}^T\mathbf{K}\mathbf{v} - \mathbf{f}^T\mathbf{v} + \boldsymbol{\nu}^T(\mathbf{B}\mathbf{v} - \mathbf{c})$$

and problem (3.1.1) is equivalent to the saddle point problem

$$L(\mathbf{u}, \boldsymbol{\lambda}) = \max_{\boldsymbol{\nu}\in\Lambda} \min_{\mathbf{v}\in\mathbb{R}^{\mathtt{n}}} L(\mathbf{v}, \boldsymbol{\nu}). \tag{3.1.5}$$

Let us consider the inner problem for any fixed $\boldsymbol{\nu} \in \Lambda$,

$$\text{find} \ \ \mathbf{u}_\nu \in \mathbb{R}^{\mathtt{n}} \quad : \quad L(\mathbf{u}_\nu, \boldsymbol{\nu}) \leq L(\mathbf{v}, \boldsymbol{\nu}) \qquad \forall \mathbf{v} \in \mathbb{R}^{\mathtt{n}}. \tag{3.1.6}$$

Since the function $L(\cdot, \boldsymbol{\nu})$ is convex for any $\boldsymbol{\nu} \in \Lambda$, the problem (3.1.6) is equivalent to

$$\mathbf{K}\mathbf{u}_\nu - \mathbf{f} + \mathbf{B}^T\boldsymbol{\nu} = \mathbf{o}. \tag{3.1.7}$$

Equation (3.1.7) has a solution if and only if

$$\mathbf{f} - \mathbf{B}^T\boldsymbol{\nu} \in Im\,\mathbf{K}, \tag{3.1.8}$$

which can be expressed more conveniently by means of a matrix $\mathbf{R} \in \mathbb{R}^{\mathtt{n}\times l}$ whose columns span the null space of $\mathbf{K}$ as

$$\mathbf{R}^T(\mathbf{f} - \mathbf{B}^T\boldsymbol{\nu}) = \mathbf{o}. \tag{3.1.9}$$

A solution $\mathbf{u}_\nu$ of (3.1.7) may be expressed as

$$\mathbf{u}_\nu = \mathbf{K}^\dagger(\mathbf{f} - \mathbf{B}^T\boldsymbol{\nu}) + \mathbf{R}\boldsymbol{\alpha}_\nu, \tag{3.1.10}$$

where $\mathbf{K}^\dagger$ is a generalized inverse to $\mathbf{K}$ and $\boldsymbol{\alpha}_\nu \in \mathbb{R}^l$.

After substituting expression (3.1.10) into problem (3.1.5), changing the signs, and omitting the constant term, we get that $\boldsymbol{\lambda}$ solves the minimization problem

$$\min\ \Theta(\boldsymbol{\nu}) \quad \text{s.t.} \quad \boldsymbol{\nu} \in \Lambda \quad \text{and} \quad \mathbf{R}^T(\mathbf{f} - \mathbf{B}^T\boldsymbol{\nu}) = \mathbf{o}, \tag{3.1.11}$$

where

$$\Theta(\boldsymbol{\nu}) = \frac{1}{2}\boldsymbol{\nu}^T\mathbf{B}\mathbf{K}^\dagger\mathbf{B}^T\boldsymbol{\nu} - \boldsymbol{\nu}^T(\mathbf{B}\mathbf{K}^\dagger\mathbf{f} - \mathbf{c}). \tag{3.1.12}$$

Once the solution $\boldsymbol{\lambda}$ of (3.1.11) is known, the solution $\mathbf{u} \equiv \mathbf{u}_\lambda$ of (3.1.1) may be evaluated by (3.1.10) with

$$\boldsymbol{\alpha} \equiv \boldsymbol{\alpha}_\lambda = (\mathbf{R}^T\overline{\mathbf{B}}^T\overline{\mathbf{B}}\mathbf{R})^{-1}\mathbf{R}^T\overline{\mathbf{B}}^T(\overline{\mathbf{c}} - \overline{\mathbf{B}}\mathbf{K}^\dagger(\mathbf{f} - \mathbf{B}^T\boldsymbol{\lambda})),$$

i.e.

$$\mathbf{u} \equiv \mathbf{u}_\lambda = \mathbf{K}^\dagger(\mathbf{f} - \mathbf{B}^T\boldsymbol{\lambda}) + \mathbf{R}\boldsymbol{\alpha},$$

where the matrix $\overline{\mathbf{B}}$ and the vector $\overline{\mathbf{c}}$ are formed by the rows of $\mathbf{B}$ and $\mathbf{c}$ corresponding to all equality constraints and all active inequality constraints.

3.2 Preconditioning by the projectors to the rigid body modes

Even though problem (3.1.11) is much more suitable for computations than (3.1.1) we can improve it by preconditioning based on orthogonal projectors. Let us denote

$$\begin{aligned} \mathbf{F} &= \mathbf{B}\mathbf{K}^\dagger\mathbf{B}^T, & \mathbf{d} &= \mathbf{B}\mathbf{K}^\dagger\mathbf{f} - \mathbf{c}, \\ \mathbf{N} &= -\mathbf{R}^T\mathbf{B}^T, & \mathbf{e} &= -\mathbf{R}^T\mathbf{f}. \end{aligned}$$

Now we can introduce preconditioned minimization problem using orthogonal projectors as follows

$$\min \quad \frac{1}{2}\boldsymbol{\lambda}^T(\mathbf{P}\mathbf{F}\mathbf{P}+\rho\mathbf{Q})\boldsymbol{\lambda}-\boldsymbol{\lambda}^T\mathbf{P}(\mathbf{d}-\mathbf{F}\boldsymbol{\lambda}_{Im}) \quad \text{s.t.} \quad \mathbf{N}\boldsymbol{\lambda} = \mathbf{o} \quad \text{and} \quad \boldsymbol{\lambda} \in \Lambda - \boldsymbol{\lambda}_{Im}, \tag{3.2.13}$$

where ρ is an arbitrary positive constant, $\boldsymbol{\lambda}_{Im} = \mathbf{N}^T(\mathbf{N}\mathbf{N}^T)^{-1}\mathbf{e}$, and

$$\mathbf{Q} = \mathbf{N}^T(\mathbf{N}\mathbf{N}^T)^{-1}\mathbf{N} \qquad \text{and} \qquad \mathbf{P} = \mathbf{I} - \mathbf{Q}$$

denote the orthogonal projectors onto the image space of $\mathbf{N}^T$ and onto the kernel of $\mathbf{N}$, respectively. Here $\boldsymbol{\lambda}_{Im}$ is used to homogenize equality constraints $\mathbf{N}\boldsymbol{\lambda} = \mathbf{e}$ which enables minimization over a subset of the vector space instead of a subset of the affine space.

The problem (3.2.13) is solved using SMALSE-M algorithm in $o(1)$ iterations.

More details about the properties and implementation of the SMALSE-M algorithm may be found in [22, 19, 25].

3.3 Numerical experiments

In this section we introduce two benchmarks, which we discuss in more details in the next chapter, where we compare elastic and elasto-plastic models. For these benchmarks we consider the stopping tolerance of the SMALSE-M algorithm as

$$\epsilon_{SMALSE} = 10^{-7}. \tag{3.3.14}$$

3.3.1 2D Hertz problem with two different materials

We consider the contact problem of the Hertz type in 2D between two elastic bodies. The geometry of the problem is depicted in Figure 3.2. The bodies Ω^1, Ω^2 are 100 x 100 and the lower boundary of Ω^1 is curved. The body Ω^1 is free in y direction. On the boundaries Γ_U^1 and Γ_U^2 we impose the symmetry conditions. The indicated traction forces are prescribed by the function $g(t) = 50$. The regular mesh is generated in MatSol and has 3 362 nodes and 6 400 triangles. We decompose the bodies Ω^1, Ω^2 into 32 subdomains (see Figure 3.3). After decomposition we have 7 744 primal variables and 1 143 dual variables. The bodies Ω^1, Ω^2 are made of homogenous isotropic materials with the parameters $E^1 = 70\,000$, $E^2 = 210\,000$, $\nu^1 = 0.35$, and $\nu^2 = 0.3$.

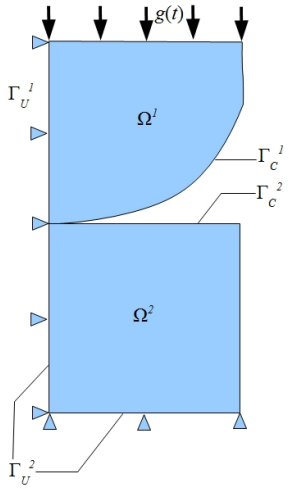

Figure 3.2: Geometry

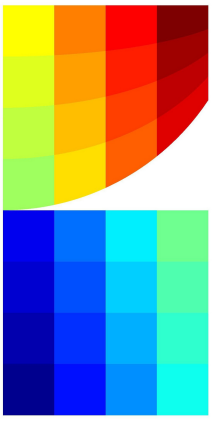

Figure 3.3: Domain decomposition into 32 subdomains

The problem is solved by SMALSE-M implemented in MatSol. Distribution of the von Mises stress $\|\mathrm{dev}(\sigma)\|_F$ and the total displacement $\|u\|$ are depicted in Figures 3.4 and 3.5, respectively.

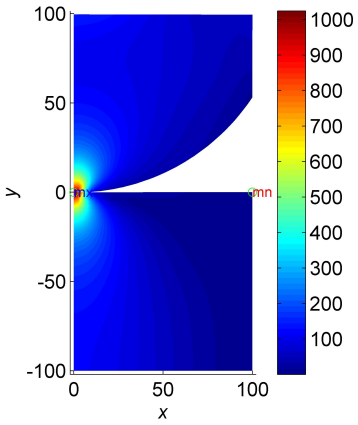

Figure 3.4: Distribution of von Mises stress

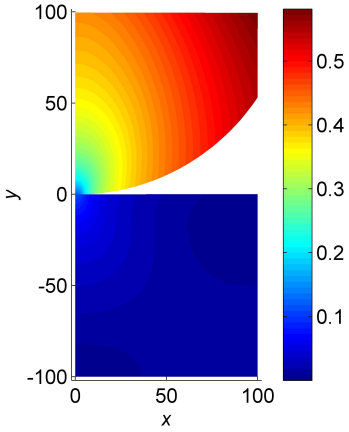

Figure 3.5: Total displacement

3.3.2 3D Hertz problem

In this benchmark, we consider the contact problem of the Hertz type in 3D between two elastic bodies. The geometry of the problem is depicted in Figure 3.6. The bodies Ω^1, Ω^2 are $5 \times 5 \times 5$ and the lower boundary of Ω^1 is curved with radius 20. The body Ω^1 is free in z direction. On the boundaries Γ_U^1 and Γ_U^2 we impose the symmetry conditions. The indicated traction forces are prescribed by the function $g(t) = 100$. The mesh is generated in MatSol again and has 3 456 nodes and 2 000 hexahedrons. Finally, we decompose the bodies Ω^1, Ω^2 into 16 subdomains (see Figure 3.7). After decomposition we have 10 368 primal variables and 2 987 dual variables. The bodies Ω^1, Ω^2 are made of homogenous isotropic materials with the parameters $E^1 = E^2 = 206\,900$ and $\nu^1 = \nu^2 = 0.29$.

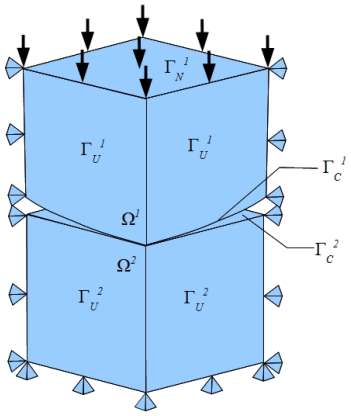

Figure 3.6: Geometry

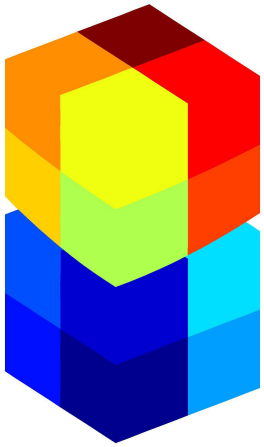

Figure 3.7: Domain decomposition into 16 subdomains

The problem is solved by SMALSE-M implemented in MatSol. Distribution of the von Mises stress $\|\mathrm{dev}(\sigma)\|_F$ and the total displacement $\|u\|$ are depicted in Figures 3.8

and 3.9, respectively.

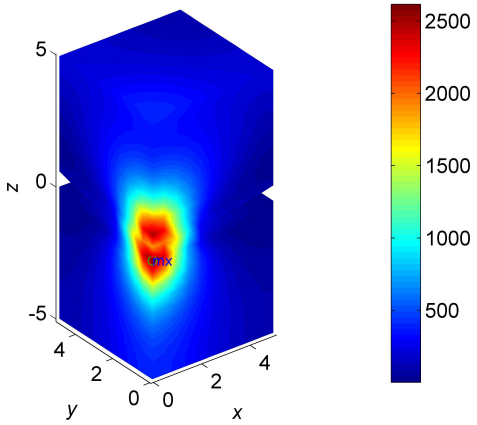

Figure 3.8: Distribution of von Mises stress

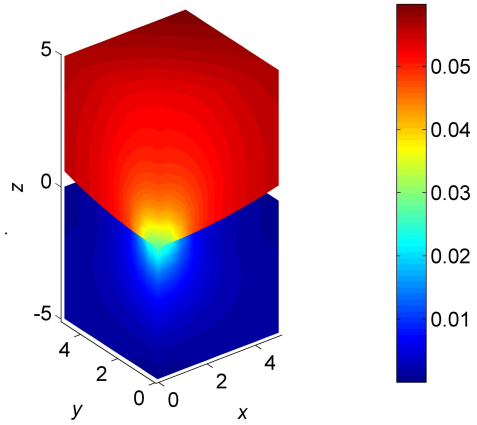

Figure 3.9: Total displacement

Chapter 4

Elasto-plasticity with contact

This chapter combines the previously mentioned algorithms to solve multibody contact problems of elasto-plasticity. We start from the TFETI based algebraic formulation of the contact problem and describe differences between elastic and elasto-plastic models. The theoretical background can be found for example in [31] and [63]. The performance of the resulting algorithm is demonstrated on numerical examples.

4.1 Algorithm for elasto-plastic problem with contact

The algorithm which we introduce below is very similar to Algorithm 4 introduced in Chapter 2 and is based on Newton-like method combined with the TFETI domain decomposition method. The main differences are:

- In each time step we solve the minimization problem with non quadratic functional and with equality and inequality constraints.
- In each Newton iteration we have to solve minimization problem with quadratic functional and with equality and inequality constraints by the algorithm SMALSE-M.
- Let us recall that in Algorithm 4 we considered three levels of displacement vectors $\mathbf{u}_k$, $\Delta\mathbf{u}_{k+1,i}$ and $\boldsymbol{\delta}\mathbf{u}_i$, belonging to the space $\mathbf{V}$. Now the displacements $\mathbf{u}_k$, $\mathbf{u}_{k+1}$ lie in the space $\mathcal{K}$ (3.1.2), the increment of displacement $\Delta\mathbf{u}_{k+1,i}$ in $\mathcal{K}_k$, and $\boldsymbol{\delta}\mathbf{u}_i$ in $\mathcal{K}_{k,i}$ where the sets $\mathcal{K}_k$ and $\mathcal{K}_{k,i}$ are defined as follows

$$\mathcal{K}_k := \mathcal{K} - \mathbf{u}_k, \quad \mathcal{K}_{k,i} := \mathcal{K} - \mathbf{u}_k - \Delta\mathbf{u}_{k+1,i-1} = \mathcal{K}_k - \Delta\mathbf{u}_{k+1,i-1}$$

  It means that

$$\mathcal{K}_{k,i} = \{\boldsymbol{\delta}\mathbf{v} \in \mathbb{R}^{\mathtt{n}} \;\mid\; \mathbf{B}_I\boldsymbol{\delta}\mathbf{v} \le \mathbf{c}_{I,i,k} \;,\; \mathbf{B}_E\boldsymbol{\delta}\mathbf{v} = \mathbf{o}\}\,,$$

  where

$$\mathbf{c}_{I,i,k} = \mathbf{c}_I - \mathbf{B}_I(\mathbf{u}_k + \Delta\mathbf{u}_{k+1,i-1}).$$

Algorithm 6 (TFETI based algorithm for solving elasto-plastic problem with contact - sequential version)**.** *Algorithm is the same as Algorithm 4 except the step 10, where the space* $\mathbf{V}$ *is replaced by* $\mathcal{K}_{k,i}$ *and PCGP by SMALSE-M.*

4.2 Numerical experiments

Let us consider the same benchmarks as in Chapter 3 and compare elastic and elasto-plastic materials models. Recall that the elasto-plastic models are naturally rate-independent but this is not true in contact problems because of changing contact zone and contact directions in each time step. The benchmark introduced in Subsection 3.3.1 we resolve for von Mises model with isotropic hardening and benchmark in Subsection 3.3.2 is resolved for von Mises model with kinematic hardening. The stopping tolerances of the Newton and the SMALSE-M algorithms are

$$\epsilon_{Newton} = 10^{-5} \text{ and } \epsilon_{SMALSE} = 10^{-9}, \tag{4.2.1}$$

respectively.

4.2.1 2D Hertz problem with von Mises criterion and isotropic hardening

Let us consider the same benchmark as in Subsection 3.3.1. The plastic material parameters are $\sigma_y^1 = 150,\ \sigma_y^2 = 450,\ H_m^1 = 10\,000$, and $H_m^2 = 100$. The indicated traction forces with the history of loading taking into account are prescribed by the function

$$g(t) = 50\sin(2\pi t),\ t \in [t_0, t^*],\ t_0 = 0,\ t^* = \frac{1}{4}. \tag{4.2.2}$$

We can consider only two variants of the equidistant time discretization but the results were so much different and that's why we consider three variants of the equidistant time discretization characterized by the time step Δt:

(a) $\Delta t = 1/4$, $N = 1$,

(b) $\Delta t = 1/16$, $N = 4$,

(c) $\Delta t = 1/24$, $N = 8$.

In Table 4.1, the number of the SMALSE-M iterations, the number of the Hessian multiplications, the number of plastic elements, and the relative convergence criterion are reported for each Newton's iteration of Algorithm 6 and the time discretization (a). In this case, we observe super linear convergence of the Newton method after identification of the plastic zone.

The computational history of Algorithm 6 for the time discretization (b) and (c), and the stopping tolerances (4.2.1) are documented in Tables 4.2 and 4.3, respectively.

Newton iter.	Number of SMALSE-M iters.	Number of Hessian multi.	Number of plastic elem.	Stopping criterion
1	87	414	0	1
2	313	1 974	210	1.5044e-1
3	361	2 305	294	2.2840e-2
4	377	2 417	314	3.2520e-3
5	432	854	315	5.4779e-5
6	418	833	315	6.7633e-7

Table 4.1: 1 time step

Time step	Number of Newton iters.	Total number of SMALSE-M iterations	Total number of Hessian multi.	Number of plastic elems.
1	4	1 099	3 201	13
2	4	1 536	6 755	65
3	5	1 674	9 516	161
4	5	1 554	9 832	303

Table 4.2: 4 time steps

Comparing Tables 4.1, 4.2, and 4.3 we see that the numbers of plastic elements are very similar but differences in the computed solutions are relatively large and well visible in Figures 4.1 - 4.3 and documented in Table 4.4. These differences result form the changing contact zone and improving contact directions in each time step.

Total displacement for the time discretization (c) is depicted in Figure 4.4. Growing of the plastic zone is documented in Figures 4.5 - 4.8 at the times $t_2,\ t_4,\ t_6$, and t_8.

4.2.2 3D Hertz problem with von Mises criterion and kinematic hardening

Let us consider the same benchmark as in Subsection 3.3.2. The plastic material parameters are $\sigma_y^1 = \sigma_y^2 = 450$ and $c_0^1 = c_0^2 = 100$. The indicated traction forces in Figure 3.6 with the history of loading taking into account are prescribed by the function

$$g(t) = 100\sin(2\pi t),\ t \in [t_0, t^*],\ t_0 = 0,\ t^* = \frac{1}{4}. \tag{4.2.3}$$

Further let us consider the same time discretizations as in the previous subsection.

In Table 4.5, the number of the SMALSE-M iterations, the number of Hessian multiplications, the number of plastic elements, and the relative convergence criterion are reported for each Newton's iteration of Algorithm 6 and the time discretization (a). In this case, we also observe super linear convergence of the Newton method after identification of the plastic zone.

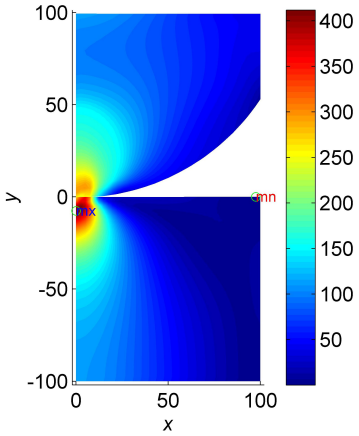

Figure 4.1: Distribution of von Mises stress for 1 time step

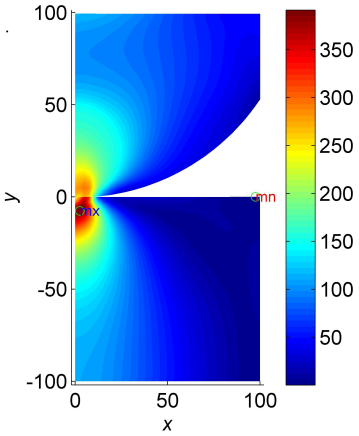

Figure 4.2: Distribution of von Mises stress for 4 time steps

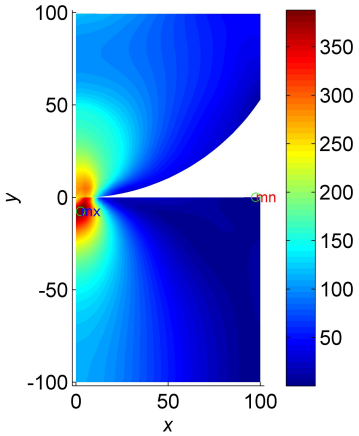

Figure 4.3: Distribution of von Mises stress for 8 time steps

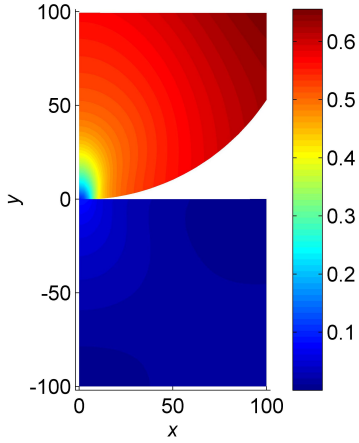

Figure 4.4: Total displacement for 8 time steps

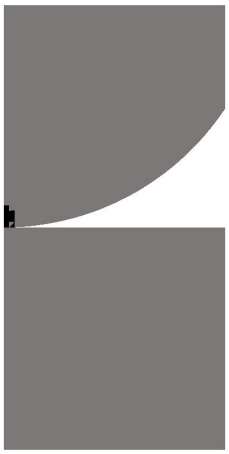

Figure 4.5: Elastic (gray color) and plastic (black color) elements at time t_2

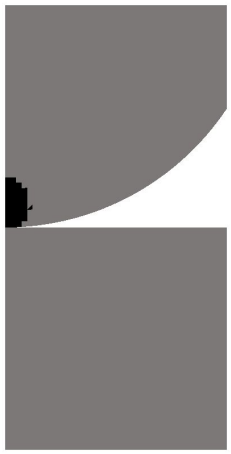

Figure 4.6: Elastic (gray color) and plastic (black color) elements at time t_4

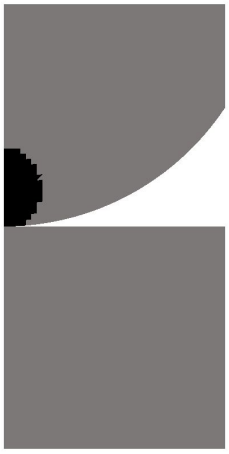

Figure 4.7: Elastic (gray color) and plastic (black color) elements at time t_6

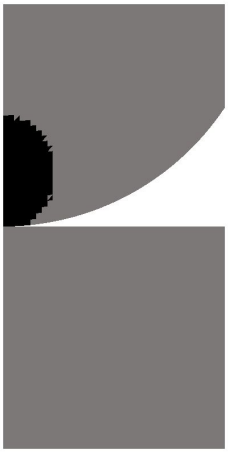

Figure 4.8: Elastic (gray color) and plastic (black color) elements at time t_8

Time step	Number of Newton iters.	Total number of SMALSE-M iterations	Total number of Hessian multi.	Number of plastic elems.
1	4	1 059	3 459	4
2	5	1 483	3 569	12
3	5	1 899	4 367	35
4	5	1 539	5 914	65
5	4	1 284	6 721	101
6	5	1 652	9 776	160
7	4	1 073	6 742	221
8	4	1 229	7 321	301

Table 4.3: 8 time step

	1 time step	4 time steps	8 time steps
max u_1	8.2669e-2	8.4316e-2	8.4862e-2
max u_2	6.6244e-1	6.5255e-1	6.5038e-1
max $\|u\|$	6.6751e-1	6.5797e-1	6.5690e-1
max $\|\mathrm{dev}(\sigma)\|_F$	4.1187e+2	3.9134e+2	3.8801e+2
min $\|\mathrm{dev}(\sigma)\|_F$	1.5435e-1	1.5449e-1	1.5458e-1

Table 4.4: Overview

The computational history of Algorithm 6 for the time discretization (b) and (c), and the stopping tolerances (4.2.1) are documented in Tables 4.6 and 4.7, respectively.

From these tables, Table 4.8, and Figures 4.9 - 4.11 we see that the final numbers of plastic elements and the corresponding solutions differ. This is similar phenomen as in the previous example.

Newton iter.	Number of SMALSE-M iters.	Number of Hessian multi.	Number of plastic elem.	Stopping criterion
1	48	447	0	1
2	9	426	585	1.7580e-1
3	10	360	1 506	3.7918e-2
4	8	319	1 722	2.6579e-2
5	8	340	1 829	2.2127e-3
6	8	344	1 845	9.2467e-5
7	8	346	1 845	5.1791e-7

Table 4.5: 1 time step

Total displacement for the time discretization (c) is depicted in Figure 4.12. Growing of the plastic zone is well documented in Figures 4.13 - 4.16 at the times t_2, t_4, t_6, and t_8.

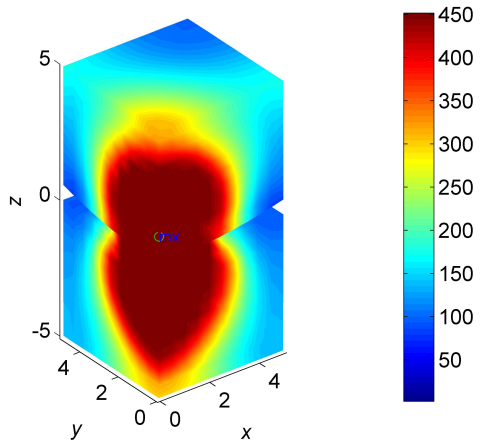

Figure 4.9: Distribution of von Mises stress for 1 time step

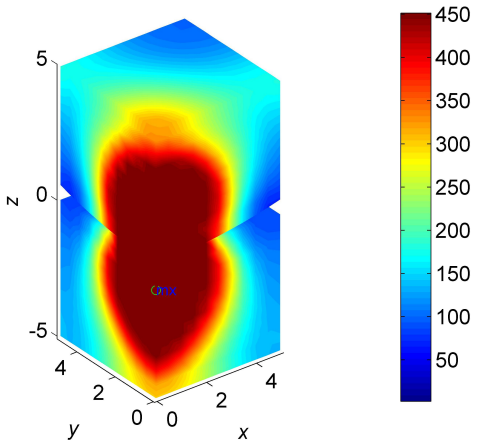

Figure 4.10: Distribution of von Mises stress for 4 time steps

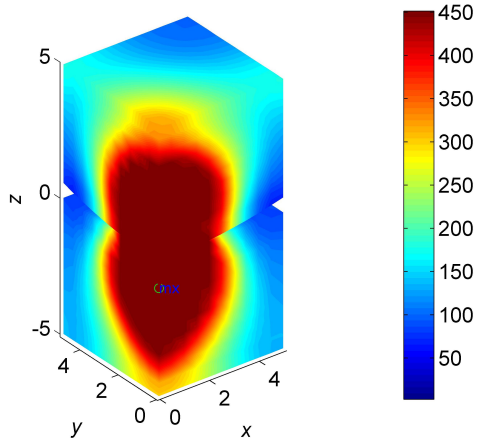

Figure 4.11: Distribution of von Mises stress for 8 time steps

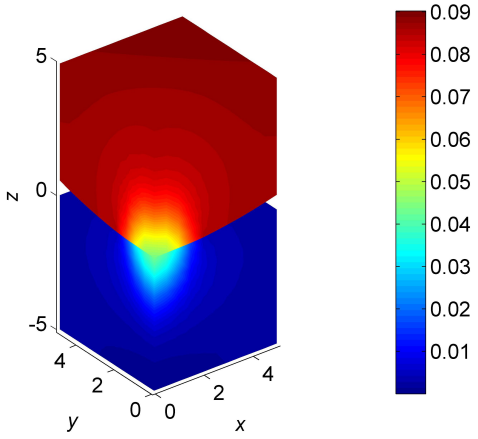

Figure 4.12: Total displacement for 8 time steps

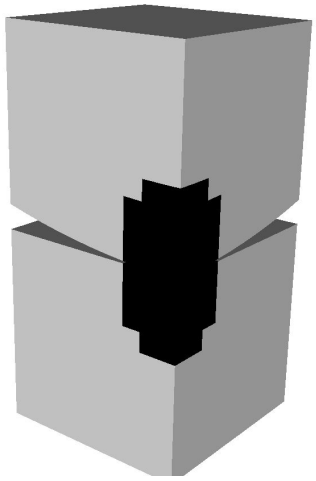

Figure 4.13: Elastic (gray color) and plastic (black color) elements at time t_2

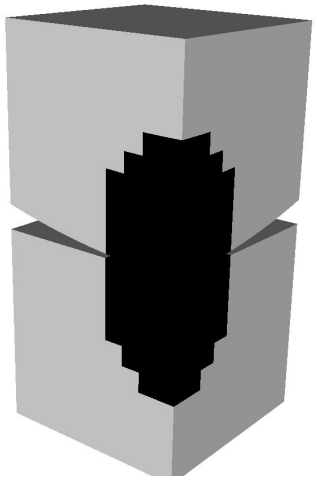

Figure 4.14: Elastic (gray color) and plastic (black color) elements at time t_4

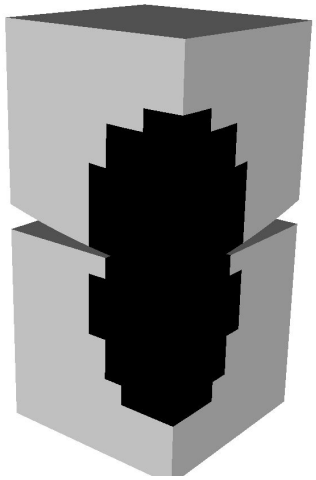

Figure 4.15: Elastic (gray color) and plastic (black color) elements at time t_6

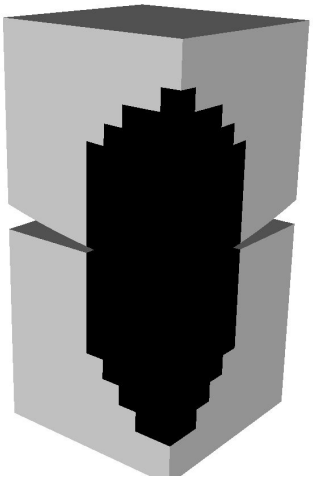

Figure 4.16: Elastic (gray color) and plastic (black color) elements at time t_8

Time step	Number of Newton iters.	Total number of SMALSE-M iterations	Total number of Hessian multi.	Number of plastic elems.
1	5	136	2 043	314
2	7	89	3 006	967
3	6	67	2 762	1 462
4	5	66	2 475	1 768

Table 4.6: 4 time steps

Time step	Number of Newton iters.	Total number of SMALSE-M iterations	Total number of Hessian multi.	Number of plastic elems.
1	6	199	2 282	116
2	5	81	1 842	314
3	6	107	3 084	626
4	5	62	2 274	941
5	5	55	2 027	1 180
6	5	63	2 440	1 440
7	5	77	2 866	1 658
8	4	72	2 538	1 754

Table 4.7: 8 time steps

4.2.3 Real world problem: yielding clamp connection

We also test our algorithms on real world problems. We consider analysis of the stress in the yielding clamp connection of steel arched supports depicted in Figire 4.17 and analyzed for the case of elastic material model in [23]. This type of construction is used to support the mining openings. It is a typical multibody contact, where the yielding connection plays the role of the mechanical protection against destruction, i.e., against the total deformation of the supporting arches. The symmetry of the problem enables to consider only a part of the clamp connection depicted in Figure 4.18. The problem is decomposed into 250 subdomains using METIS and discretized by 1,592,853 and 216,604 primal and dual variables, respectively.

We solve this benchmark in parallel using 20 cores. Because we do not prescribe any traction forces but only prestress effect in screw, we must modify Algorithm 6. We consider the time discretization with 4 time steps but in each time step we start with zero displacement approximation, i.e. we do not add stress, hardening, and displacement increments to previous values. It means that in each time step we perform only contact direction corrections. For this approach we use damping with value 0.5, to improve convergence. The behaviour may be better when we use mortars for contact condition. In Figures 4.18 and 4.19 we depict the von Mises stress distribution and total displacement, respectively.

	1 time step	4 time steps	8 time steps
max u_1	7.09386e-2	6.14744e-2	6.04612e-2
max u_2	7.09386e-2	6.14744e-2	6.04612e-2
max u_3	9.21676e-1	9.05577e-1	9.01938e-1
max $\|u\|$	9.22038e-1	9.05971e-1	9.02336e-1
max $\|\text{dev}(\sigma)\|_F$	4.51507e+2	4.51376e+2	4.51364e+2
min $\|\text{dev}(\sigma)\|_F$	1.93774e+0	2.08079e+0	2.24098e+0

Table 4.8: Overview

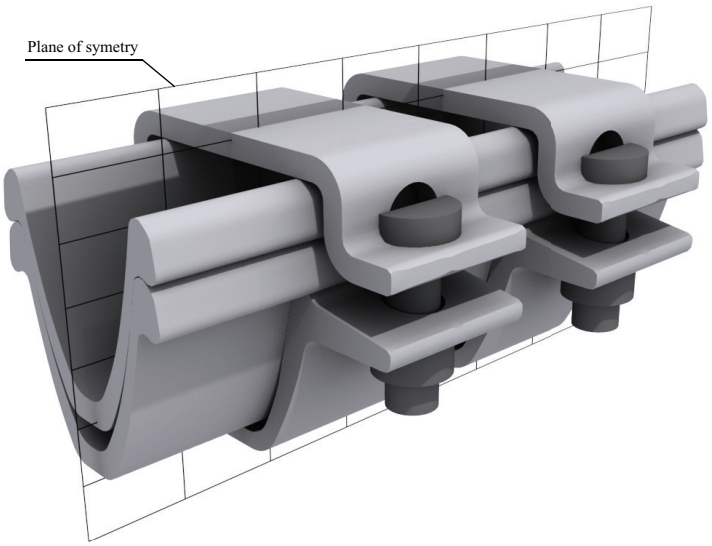

Figure 4.17: Yielding clamp connection

4.3 Parallel scalability for elasto-plastic problem with contact

In this section we demonstrate the parallel scalability for elasto-plastic problem with contact boundary condition. We assume the same geometry, boundary condition, material parametrs and traction forces as in Subsection 4.2.2.

For simplicity, we demonstrate parallel scalability for one time step problem. For the spatial discretization of Ω^1 and Ω^2, let us consider one level of hexahedron mesh generated by MatSol with 18 522 nodes and 16 000 elements decomposed into 128 subdomains. The number of primal variables is 82 944 and the number of dual variables is 29 583.

In Table 4.9 and Figure 4.20, we document the parallel scalability for different num-

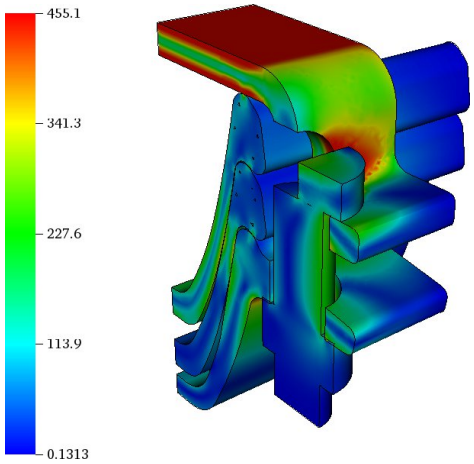

Figure 4.18: Distribution of von Mises stress

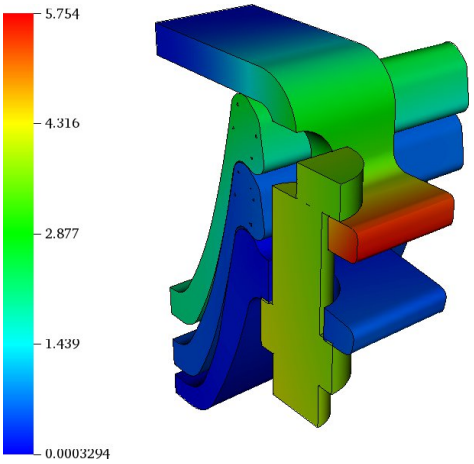

Figure 4.19: Total displacement

ber of processors. We see that the number of the Newton iterations, number of plastic elements, total number of SMALSE-M iterations, and total number of the Hessian multiplications remain constant for all cases.

Number of cores	2	3	5	9	17	33
Number of plastic elems.	14 765	14 765	14 765	14 765	14 765	14 765
Number of Newton iter.	7	7	7	7	7	7
Total number of SMALSE-M iter.	120	120	120	120	120	120
Total number of Hessian multi.	19 357	19 357	19 357	19 357	19 357	19 357
Total time [sec]	4 038.0	2 575.2	1 865.1	1 322.5	1 030.1	872.4

Table 4.9: Paralel scalability of Algorithm 4 for different number of cores in 3D.

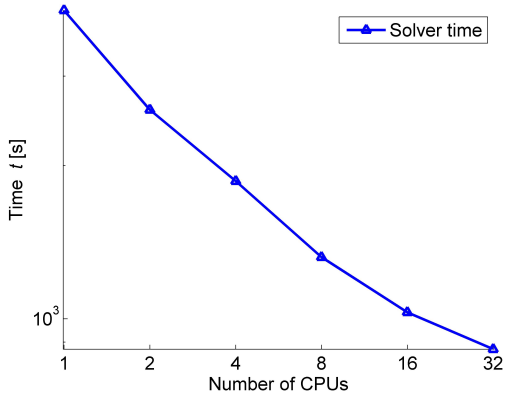

Figure 4.20: Parallel scalability (3D case)

Conclusion

In the thesis we combined the TFETI based domain decomposition method with non-smooth Newton method so that they can be used to solve selected problems of elasto-plasticity. Concretely we considered three elasto-plastic models: von Mises model with isotropic hardening, von Mises model with kinematic hardening and Drucker-Prager model with perfect plasticity. By this we extended capabilities of MatSol library to solve of selected elasto-plastic problem. The proposed algorithms were implemented in MatSol library in both sequential and parallel versions for several types of finite elements. Convergence properties, effectivity and scalability of the proposed algorithms were illustrated on the numerical examples. During the work we encountered several engineering, mathematical and numerical problems. Let us name some of them:

1. The influence of inaccuracy of the solvers based on TFETI on the convergence of the Newton method. We studied this issues only on a level of numerical examples. We analyzed the dependance of the inner solver accuracy on the accuracy of the Newton method to keep a locally quadratic convergence.

2. Work with linear simplicial elements. However the MatSol implementation was performed for e.g. hexahedrons in combination with a suitable numerical quadrature.

3. A solution of the perfectly plastic problem in displacement. From a theoretical point of view it is known that the mathematical formulation of the problem of perfect plasticity is in some respect disputable. Sobolev spaces do not seem to be sufficient for the formulation of such problem - we have no information about the uniqueness of a solution and the conditions for an existence of spatialy discretized solutions are hard to confirm. Despite this, problems of perfect plasticity are in engineering usually solved by FEM in displacement although the conditions for a convergence of Newton method may not be fulfilled in general. In the work we dealt with convergence problems by considering so called elastic stiffness matrix instead of tangential stiffness matrix in each Newton interation. Thus we obtained a more stable but not quadratic convergence.

4. Combination of elasto-plasticity with contact boundary conditions. This combination is not yet described in more detail in existing literature. In our work we extended existing TFETI solvers for elasticity with contact to contact problems of elasto-plasticity. Therefore we tried to generalize Newton method for the solution of non-linear variational inequalities. The numerical experiment show that even in this case we can observe a locally quadratic convergence. A theoretical background for this problems will be given in the future works.

Bibliography

[1] Alberty J., Carstensen C., Zarrabi D.: *Adaptive numerical analysis in primal elasto-plasticity with hardening*, Comput. Methods Appl. Mech. Engrg. 171, 175-–204, 1999.

[2] Alberty, J., Carstensen, C., Funken, S.A., Klose, R.: *Matlab implementation of the finite element method in elasticity* Computing (Vienna/New York) 69 (3) , 239–263, 2002.

[3] Andersson, J.C., Martin, C.D.: *The Äspö pillar stability experiment: Part I-Experiment design*, International Journal of Rock Mechanics and Mining Science 46, 865–878, 2009.

[4] Andersson, J.C., Martin, C.D., Stille, H.: *The Äspö Pillar Stability Experiment: Part II-Rock mass response to coupled excavation-induced and thermal-induced stresses*, International Journal of Rock Mechanics and Mining Science 46, 879–895, 2009.

[5] Avery, P., Farhat, C.: *The FETI family of domain decomposition methods for inequality-constrained quadratic programming: Application to contact problems with conforming and nonconforming interfaces*, Computer Methods in Applied Mechanics and Engineering, 198(21-26):1673-1683, 2009.

[6] Badea, L., Gilormini, P.: *Application of a domain decomposition method to elasto-plastic problems*, International Journal of Solids and Structures 31 (5), Pages 643–656, 1994.

[7] Bartels, S., Mielke, A., Roubíček, T.: *Quasistatic small-strain plasticity in the limit of vanishing hardening and its numerical approximation*, SIAM J. Numer. Anal., to appear.

[8] Bazaraa, MS., Shetty, CM., Sherali, HD.: *Nonlinear Programming, Theory and Algorithms. Second Edition*, Wiley, New York, 1993.

[9] Benzi, M., Golub, G. H., Liesen, J.: *Numerical solution of saddle point problems*, Acta Numerica, Cambridge University Press, 1–137, 2005.

[10] Blaheta, R.: *Numerical methods in elasto-plasticity*, Documenta Geonica 1998, PERES Publishers, Prague, 1999.

[11] Blaheta, R., Byczanski, P., Čermák, M., Hrtus, R., Kohut, R., Kolcun, A., Malík, J., Sysala, S.: *Analysis of Äspö Pillar Stability Experiment: Continuous TM model development and calibration*, Submitted.

[12] Brokate, M., Sprekels J.: *Hysteresis and Phase Transitions*, Springer-Verlag New York, 1996.

[13] Brzobohatý, T., Dostál, Z., Kovář, P., Kozubek, T. and Markopoulos, A.: *Cholesky decomposition with fixing nodes to stable evaluation of a generalized inverse of the stiffness matrix of a floating structure*, International Journal for Numerical Methods in Engineering 88 (5), 493–509, 2011.

[14] Carstensen C.: *Nonlinear Interface Problems in Solid Mechanics: Finite El- ement and Boundary Element Couplings*, habilitation thesis, University Hannover, 1993.

[15] Carstensen, C., Klose, R.: *Elastoviscoplastic finite element analysis in 100 lines of Matlab* Journal of Numerical Mathematics 10 (3), 157–192, 2002.

[16] Clarke, H. F.: *Optimization and Nonsmooth Analysis*, Wiley, New York, 1983.

[17] Čermák, M., Kozubek, T., Sysala, S., Valdman, J.: *A TFETI Domain Decomposition Solver for Elastoplastic Problems*, arXiv:1205.1926v1 [math.NA].

[18] Dostál, Z.: *Inexact Semimonotonic Augmented Lagrangians with Optimal Feasibility Convergence for Convex Bound and Equality Constrained Quadratic Programming*, SIAM J. Numer. Anal. 43, 96–115, 2005.

[19] Dostál, Z.: *Optimal Quadratic Programming Algorithms, with Applications to Variational Inequalities*, 1st edition, SOIA 23, Springer US, New York, 2009.

[20] Dostál, Z., Gomes, FAM., Santos, SA.: *Solution of contact problems by FETI domain decomposition with natural coarse space projection*, Comput. Methods Appl. Mech. Eng. 190, 13–14, 1611–1627, 2000.

[21] Dostál, Z., Horák, D., Kučera, R.: *Total FETI - an easier implementable variant of the FETI method for numerical solution of elliptic PDE*, Communications in Numerical Methods in Engineering 22 (12), 1155–1162, 2006.

[22] Dostál, Z., Kozubek, T.: *An optimal algorithm and superrelaxation for minimization of a quadratic function subject to separable convex constraints with applications*, accepted for publishing in Mathematical Programming, 2011. DOI: 10.1007/s10107-011-0454-2

[23] Dostál, Z., Kozubek, T., Markopoulos, A., Brzobohatý, T., Vondrák, V., Horyl, P.: *Theoretically supported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction*, CMAME 205–208, 110—120, 2012.

[24] Dostál, Z., Kozubek, T., Vondrák, V., Brzobohatý, T., Markopoulos, A.: *Scalable TFETI algorithm for the solution of multibody contact problems of elasticity*, International Journal for Numerical Methods in Engineering 41, 675–696, 2010.

[25] Dostál, Z., Kučera, R.: *An optimal algorithm for minimization of quadratic functions with bounded spectrum subject to separable convex inequality and linear equality constraints*, SIAM Journal on Optimization 20 (6), 2913—2938, 2010.

[26] Dostál, Z., Schöberl, J.: *Minimizing quadratic functions over non-negative cone with the rate of convergence and finite termination*, Comput. Optim. Appl., 30 (1), 23–44, 2005.

[27] Dureisseix, D., Farhat, C.: *A numerically scalable domain decomposition method for solution of frictionless contact problems*, Int. J. Numer. Methods Eng. 50, 12, 2643–2666, 2001.

[28] Farhat, C., Roux, F-X.: *An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems*, SIAM J. Sci. Stat. Comput. 13, 379–396, 1992.

[29] Farhat, C., Mandel, J., Roux, F-X.: *Optimal convergence properties of the FETI domain decomposition method*, Comput. Methods Appl. Mech. Eng. 115, 365–385, 1994.

[30] Farhat, C., Géradin, M.: *On the general solution by a direct method of a large scale singular system of linear equations: application to the analysis of floating structures*, Int. J. Num. Met. Eng. 41, 675–696, 1997.

[31] Fortin, M., and Glowinski, R.: *Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary Value Problems*, Studies in Mathematics and its Applications 15, North-Holland, 1983.

[32] Fučík, S., Kufner, A.: *Nonlinear Differential Equation*, Elsevier, Amsterdam, 1980.

[33] Gowda, M. S.: *Inverse and implicit function theorem for H-Differentable and semismooth function*, Optimization methods and Software 19, 443–461, 2004.

[34] Gruber P., Kienesberger J., Langer U., Schoeberl J., Valdman J.: *Fast solvers and a posteriori error estimates in elastoplasticity*, U. Langer and P. Paule (editors), Numerical and Symbolic Scientific Computation: Progress and Prospects, Texts and Monographs in Symbolic Computation, Wien: Springer, 45–63, 2011.

[35] Gruber, P. G., Valdman, J.: *Solution of One-Time Step Problems in Elastoplasticity by a Slant Newton Method*, SIAM J. Sci. Comp. 31, 1558–1580, 2009.

[36] Han, W., Reddy, B. D.: *Plasticity, Mathematical Theory and Numerical Analysis*, Springer-Verlag, New York, 1999.

[37] Haslinger, J., Dostál, Z., Kučera, R.: *On the splitting type algorithm for the numerical realization of the contact problems with Coulomb friction*, Comput. Methods. Appl. Mech. Engrg., 191, 2261—2281, 2002.

[38] Haslinger, J., Hlavacek, I., Necas, J.: *Numerical Methods for Unilateral Problems in Solid Mechanics*, In CIARLET, P. G.; LIONS, J. L. Handbook of Numerical Analysis : Volume IV.. Amsterdam : Elsevier, pp. 313-485, 1996.

[39] Haslinger, J., Kučera, R., Dostál, Z.: *An algorithm for the numerical realization of 3D contact problems with Coulomb friction*, J. Comput. Appl. Math., 164–165, 387–408, 2004.

[40] Johnson C.: *Existence theorems for plasticity problems*, J. Math. pures et appl. 55, 431-–444, 1976.

[41] Johnson C.: *On plasticity with hardening*, J. Math. Anal. Appls. 62, 325—336, 1978.

[42] Kienesberger J., Langer U., Valdman J.: *On a robust multigrid-preconditioned solver for incremental plasticity problems*, Proceedings of IMET 2004 - Iterative Methods, Preconditioning & Numerical PDEs, Prague.

[43] Kikuchi, N., Oden, JT.: *Contact Problems in Elasticity*, SIAM, Philadelphia, 1988.

[44] Korneev V.G., Langer U.: *Approximate solution of plastic flow theory problems*, Teubner-Verlag, Leipzig, volume 69, 1984.

[45] Kozubek, T., Markopoulos, A., Brzobohatý, T., Kučera, R., Vondrák, V., Dostál, Z.: *MatSol - MATLAB efficient solvers for problems in engineering*, http://matsol.vsb.cz/.

[46] Kozubek, T., Vondrák, V., Menšík M., Horák D., Dostál, Z., Hapla V., Kabelíková P., Čermák M.: *Total FETI domain decomposition method and its massively parallel implementation*, Advances in Engineering Software, accepted 2012.

[47] Krejčí P.: *Hysteresis, Convexity and Dissipation in Hyperbolic Equations*, GAKUTO International Series, Mathematical Sciences and Applications, 1996.

[48] Kučera, R., Kozubek, T., Markopoulos, A.: *On large-scale generalized inverses in solving two-by-two block linear systems*, Submitted to Linear Algebra and Its Applications, 2012.

[49] Mifflin, R.: *Semismoothness and semiconvex function in constraint optimization*, SIAM Journal on Control and Optimization 15, 957–972, 1977.

[50] Nečas, J., Hlaváček, I.: *Mathematical Theory of Elastic and Elasto-Plastic Bodies. An Introduction*, Elsevier, Amsterdam, 1981.

[51] Neto, EA S., Perić, D., Owen DRJ.: *Computational methods for plasticity, Theory and applications*, John Wiley & Sons Ltd 2008.

[52] Pang, J.-S., Sun, D., Sun, J.: *Semismoothness Homeomorfisms and Strong Stability of Semidefinite and Lorenz Cone Complementarity Problems.* Mathematics of Operational Research 28, 39–63, 2003.

[53] Qi, L., Sun, J.: *A nonsmooth version of Newton's method*, Mathematical Programming, 58, 353–367, 1993.

[54] Roux, F-X.: *Spectral analysis of interface operator*, Proceedings of the 5th Int. Symp. on Domain Decomposition Methods for Partial Differential Equations, ed. D. E. Keyes et al., SIAM, 73–90, Philadelphia 1992.

[55] Roux, F-X., Farhat, C.: *Parallel implementation of direct solution strategies for the coarse grid solvers in 2-level FETI method*, Contemporary Math. 218, 158–173, 1998.

[56] Sauter M., Wieners C.: *On the superlinear convergence in computational elasto-plasticity*, Comp. Meth. Eng. Mech. 200, 3646–3658, 2011.

[57] Simo, J. C., Hughes, T. J. R.: *Computational Inelasticity*, Interdiscip. Appl. Math. 7, Springer-Verlag, New York, 1998.

[58] Sun, D.: *A further result on an implicit function theorem for locally Lipschitz function*, Operation Research letters 28, 193–198, 2001.

[59] Suquet, P.M.: *Sur les équations de la plasticité: existence et régularité des solutions*, (in French) J. Mec., 20, 3–39, 1981. Zbl 0474.73030

[60] Stein, E. et al.: *Error-Controlled Adaptive Finite Elements in Solid Mechanics*, Wiley, Chichester, UK, 2003.

[61] Sysala, S.: *Application of a modified semismooth Newton method to some elasto-plastic problems*, Math. Comput. Simul., 2012, http://dx.doi.org/10.1016/j.matcom.2012.03.012, ARTICLE IN PRESS.

[62] Sysala, S.: *Properties and simplifications of constitutive time-discretized elastoplastic operators.* Submitted to ZAMM journal in 2012.

[63] Temam, R.: *Mathmatical problems in plasticity* BORDAS, Paris, 1985.

[64] Valdman, J.: *Mathematical and Numerical Analysis of Elastoplastic Material with Multi-Surface Stress-Strain Relation*, Kiel, Germany, 2001 submitted, 2002 published at dissertation.de (ISBN 3-89825-501-8).

[65] Wieners, C.: *Multigrid methods for finite elements and the application to solid mechanics. Theorie und Numerik der Prandtl-Reuß Plastizität*, habilitation thesis, University Heidelberg, 2000.

[66] Wieners, C.: *A geometric data structure for parallel finite elements and the application to multigrid methods with block smoothing*, Computing and Visualization in Science 13 (4), 161–175, 2010.

[67] Wriggers, P.: *Contact Mechanics*, Springer, Berlin, 2005.

Author's bibliography

Proceedings international

[68] Blaheta, R., Byczanski, P., Čermák, M., Hrtus, R., Kohut, R., Kolcun, A., Malík, J., Sysala, S.: *Analysis of Äspö Pillar Stability Experiment: Continuous TM model development and calibration*, Submitted.

[69] Čermák, M., Kozubek, T., Markopoulos, A.: *An efficient parallel solver for elasto-plastic problems of mechanics*, PARENG 2011.

[70] Čermák, M., Kozubek, T., Markopoulos, A.: *An efficient FETI based solver for elasto-plastic problems of mechanics*, Computational Plasticity XI - Fundamentals and Applications, COMPLAS XI, 1330–1341, 2011, *Scopus.*

[71] Čermák, M., Kozubek, T., Markopoulos, A.: *An efficient FETI based solver for elasto-plastic problems of mechanics*, ICNAAM-2011 Conference, AIP Conference Proceedings, Volume 1389, 1782–1785, 2011, *Scopus, WoS.*

[72] Čermák, M., Kozubek, T.: *An efficient TFETI based solver for elasto-plastic problems of mechanics*, AEEE 10(1), 57–62, 2012.

[73] Čermák, M., Kozubek, T., Sysala, S., Valdman, J.: *A TFETI Domain Decomposition Solver for Elastoplastic Problems*, arXiv:1205.1926v1 [math.NA]. *Paper was sent to the Applied Mathematics and Computation, journal with the impact factor.*

[74] Kozubek, T., Vondrák, V., Menšík M., Horák D., Dostál, Z., Hapla V., Kabelíková P., Čermák M.: *Total FETI domain decomposition method and its massively parallel implementation*, Advances in Engineering Software, accepted 2012, *journal with the impact factor.*

Proceedings domestic

[75] Čermák, M., Brzobohatý, T., Markopoulos, A.: *Scalable TFETI algorithm for two dimensional multibody contact problems with friction*, Proceedings of papers, ČVUT Praha 2009.

[76] Čermák, M.: *Newton method for elasto-plasticity problems*, Proceedings of papers, ČVUT Praha 2010.

[77] Čermák, M.: *Theoretically supported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction*, WOFEX 2010 - PhD Workshop, Proceedings of papers, VSB-TU Ostrava 2010.

[78] Čermák, M., Kozubek, T., Markopoulos, A.: *An efficient solution of elasto-plastic problems in mechanics*, Seminar on Numerical Analysis & Winter School , Proceedings of the conference SNA'11, 2011.

[79] Čermák, M.: *Efficient TFETI based solver for elasto-plastic problems of mechanics*, WOFEX 2011 - PhD Workshop, Proceedings of papers, VSB-TU Ostrava 2011.

[80] Čermák, M., Kozubek, T.: *An eficient solution of elasto-plastic contact problems in mechanics*, Seminar on Numerical Analysis & Winter School , Proceedings of the conference SNA'12, 2012.

[81] Čermák, M., Sysala, S.: *Continual approach to modelling of a pillar failure process*, Seminar on Numerical Analysis & Winter School , Proceedings of the conference SNA'12, 2012.